processes

MDPI

*Article*

# Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization

**Nadikatla Chandrasekhar and Samineni Peddakrishna ***

School of Electronics Engineering, VIT-AP University, Amaravati 522237, India
* Correspondence: krishna.samineni@gmail.com

**Abstract:** In the medical domain, early identification of cardiovascular issues poses a significant challenge. This study enhances heart disease prediction accuracy using machine learning techniques. Six algorithms (random forest, K-nearest neighbor, logistic regression, Naïve Bayes, gradient boosting, and AdaBoost classifier) are utilized, with datasets from the Cleveland and IEEE Dataport. Optimizing model accuracy, GridsearchCV, and five-fold cross-validation are employed. In the Cleveland dataset, logistic regression surpassed others with 90.16% accuracy, while AdaBoost excelled in the IEEE Dataport dataset, achieving 90% accuracy. A soft voting ensemble classifier combining all six algorithms further enhanced accuracy, resulting in a 93.44% accuracy for the Cleveland dataset and 95% for the IEEE Dataport dataset. This surpassed the performance of the logistic regression and AdaBoost classifiers on both datasets. This study's novelty lies in the use of GridSearchCV with five-fold cross-validation for hyperparameter optimization, determining the best parameters for the model, and assessing performance using accuracy and negative log loss metrics. This study also examined accuracy loss for each fold to evaluate the model's performance on both benchmark datasets. The soft voting ensemble classifier approach improved accuracies on both datasets and, when compared to existing heart disease prediction studies, this method notably exceeded their results.

**Keywords:** heart disease prediction; machine learning; soft voting ensemble classifier; performance matrices



**Citation:** Chandrasekhar, N.; Peddakrishna, S. Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization. *Processes* **2023**, *11*, 1210. https://doi.org/10.3390/pr11041210

Academic Editors: Kelvin K.L. Wong, Dhanjoo N. Ghista, Andrew W.H. Ip and Wenjun (Chris) Zhang

Received: 5 March 2023
Revised: 27 March 2023
Accepted: 11 April 2023
Published: 14 April 2023



## 1. Introduction

Statistics from the World Health Organization (WHO) indicate that heart disease is a major threat to humans worldwide [1]. Heart disease can be caused by many different things, including high blood pressure, obesity, excessive cholesterol, smoking, unhealthy eating habits, diabetes, and abnormal heart rhythms [2]. Most patients die from heart disease as a result of an inadequate diagnosis at the initial phase. Therefore, it is imperative to use efficient disease classification and prediction algorithms to comprehend disease prediction. In contrast, it is necessary to implement a more accurate model in order to predict heart disease. An assessment of the accuracy of a model to predict heart-related diseases is based on its precision, F1 score, and recall performance. Association rules can also improve the prediction accuracy for heart disease models. The use of association rules on medical datasets produces a number of regulations. Most of these rules do not have any medical relevance. Furthermore, finding them can be time-consuming and impractical. This is due to the fact that the association rules are drawn from the available dataset rather than being based on an independent sample. Hence, to identify early-stage predictions for heart disease, search constraints are applied to actual datasets containing patients with heart disease. Using search constraints, a rule-generation algorithm has been used for the early detection of heart attacks [3]. Moreover, recent advances in healthcare technology have driven the development of machine learning (ML) systems for the prediction of human health diseases [4–6]. There have been many researchers working on the development of improved ML models. The primary objective of the ML technique is to generate computer code that can access and use current data to predict future data [7]. Additionally,

there are some tried-and-true methods for improving the accuracy of the model. These include adding more information to the dataset, treating missing and outlier values, feature selection, algorithm tuning, cross-validation, and ensembling. This paper implements GridsearchCV hyperparameter tuning and five-fold cross-validation to evaluate the model's performance on both benchmark datasets. It also employs an ensemble voting classifier to improve model accuracy, aiming to enhance ML model accuracy. This article presents the following significant work:

- This work examines and implements six major ML algorithms on the Cleveland and IEEE Dataport heart disease datasets, analyzing performance classification metrics.
- In the early phase, various ML classifier techniques, including random forest (RF), K-nearest neighbor (KNN), logistic regression (LR), Naive Bayes (NB), gradient boosting (GB), and AdaBoost (AB) were trained.
- The GridsearchCV hyperparameter tuning method with five-fold cross-validation and performance assessment using accuracy and negative log loss metrics was employed to achieve the highest level of accuracy.
- Finally, all classifiers were combined using a soft voting ensemble method in order to increase the accuracy of the model.

## 2. Literature Review

Several new research opportunities in healthcare have been enabled by advances in ML and advances in computing capabilities [8]. Various researchers have proposed ML algorithms to enhance the accuracy of disease prediction [9–11]. To refine the precision of the outcomes, much of the research has meticulously evaluated the presence of missing data in the dataset, a crucial aspect in the data preprocessing process. Gupta et al. [12] used Pearson correlation coefficients and different ML classifiers to replace missing values in the Cleveland dataset. Rani et al. [13] have investigated multiple imputations by the chained equations (MICE) method to deal with the missing values problem. In this case, missing values are imputed through a series of iterative predictive models. During each iteration, each variable in the dataset is assigned using the other variables. In another work, Jordanov et al. [14] proposed a KNN imputation method for the prediction of both continuous (average of the nearest neighbors) and categorical variables (most frequent). Another study used an LR model to classify cardiac disease with an accuracy of 87.1% after cleaning the dataset and identifying missing values at the time of preprocessing [15]. In contrast, some researchers have eliminated missing values. Based on DT, LR, and Gaussian NB algorithms, the features are reduced from 13 to 4 using feature selection method and reported an accuracy of 82.75% [16]. A hybrid random forest (RF) with the linear model was developed by Mohan et al. [17] and improved the accuracy of 297 records and 13 characteristics of the Cleveland dataset for heart disease prediction. Kodati et al. [18] tested several types of classifiers using Orange and Weka data-mining tools to predict heart disease with 297 records and 13 features.

In addition, the feature selection method plays an important role in improving the accuracy of the model. To select features, Shah et al. [19] utilized probabilistic principal component analysis (PCA). The Cleveland dataset was used by R. Perumal et al. [20] to develop LR and support vector machine (SVM) models with similar accuracy levels (87% and 85%, respectively). To train the ML classifiers, they used a dataset of 303 data instances and standardized and reduced features using PCA. In another study, a particle swarm optimization (PSO) technique was used to select features [21]. In contrast, Yekkala et al. [22] used a rough set-based feature selection method along with the RF algorithm and obtained an accuracy of 84%. Saw et al. [23] used a random search to find the best parameters to build an accurate prediction model. It was found that this approach uses LR for classification and is 87% accurate at predicting heart attacks. Other works have used both methods and predicted the accuracy using different algorithms. The model presented by Otoom et al. [24] used NB, SVMs, and available trees to achieve an accuracy of 84.5%. Vemban-

dasamy et al. [25] proposed an NB classifier for predicting heart disease and achieved an accuracy of 84.4%.

Further, to determine the optimum combination of heart disease predictors, Gazeloglu et al. [26] evaluated 18 ML models and three feature selection techniques for the Cleveland dataset of 303 instances and 13 variables. Recently, ten classifiers were trained to identify the most effective prediction models for precise prediction [27]. The most suitable attributes were identified using three methods of attribute selection, including a feature subset evaluator based on correlation, a chi-squared attribute evaluator, and a relief attribute evaluator. Furthermore, a hybrid feature selection method aimed at enhancing accuracy by incorporating RF, AB, and linear correlation was suggested by Pavithra et al. [28]. The implementation of this technique led to a 2% increase in the accuracy of the hybrid model, following the selection of 11 features through a combination of filter, wrapper, and embedded methods. To further enhance the accuracy, researchers have used the ensemble technique to combine different algorithms. The ensemble method for detecting heart disease was developed by Latha et al. [29] by combining NB, RF, multilayer perceptrons (MLP), and Bayesian networks based on majority voting (MV). They achieved an accuracy of 85.48%. It was also employed by an ensemble model with five classifiers, including a memory-based learner (MBI), an SVM, DT induction with information gain (DT-IG), NB, and DT initiation with the Gini index (DT-GI) [30]. As the datasets in the authors' study contained only pertinent attributes, there was no feature selection. A pre-processing step has been performed to eliminate outliers and missing values from the data. Tama et al. [31] developed an ensemble model to diagnose heart disease with an accuracy rate of 85.71%. The ensemble model utilized GB, RF, and extreme GB classifiers. Alqahtani et al. [32] developed an ensemble of ML and deep learning (DL) models to predict the disease with an accuracy rate of 88.70%. This study employed a total of six classification algorithms. Trigka et al. [33] developed a stacking ensemble model after applying SVM, NB, and KNN with a 10-fold cross-validation synthetic minority oversampling technique (SMOTE) in order to balance out imbalanced datasets. This study demonstrated that a stacking SMOTE with a 10-fold cross-validation achieved an accuracy of 90.9%. Another study used stochastic gradient descent classifiers, LR, and SVM to develop a model with an accuracy of 93% using multiple datasets [34]. For further improving accuracy, Cyriac et al. [35] utilized seven different machine-learning models as well as two ensemble methods (soft voting and hard voting). With this approach, the highest accuracy score was achieved at 94.2%. Another study developed a combined multiple-classifier predictive model approach for better prediction accuracy [36]. Five classifier models are combined with Cleveland and Hungarian datasets. A total of 590 data-valid instances and 13 attributes were taken into consideration. A baseline accuracy of 93% was achieved using the Weka data-mining tool.

In 2020, Manu Siddhartha created a new dataset by combining five well-known heart disease datasets—Switzerland, Cleveland, Hungary, Statlog, and Long Beach VA. This new dataset includes all the characteristics shared by the five datasets [37]. In the same dataset, Mert Ozcan et al. [38] investigated the use of a supervised ML technique known as the Classification and Regression Tree (CART) algorithm to predict the prevalence of heart disease and to extract decision rules that clarify the associations between the input and output variables. The outcomes of the investigation further ranked the heart disease influencing features based on their significance. The model's reliability was corroborated by an 87% accuracy in the prediction. Other researchers Rüstem Yilmaz et l. [39] worked to compare the predictive classification performances of ML techniques for coronary heart disease. Three distinct models using RF, LR, and SVM algorithms were developed. Hyper-parameter optimization was performed using a 10-fold repeated cross-validation approach. Model performance was assessed using various metrics. Results showed that the RF model exhibited the highest accuracy of 92.9%, specificity of 92.9%, sensitivity of 92.8%, F1 score of 92.8%, and negative predictive and positive predictive values of 92.9% and 92.8%, respectively.

In the field of predictive modeling, there is a constant pursuit to enhance the accuracy of classification and forecast models. The classification models are deployed to label data points while forecast models are used to predict future values. A suitable combination of models and features can enhance the accuracy of these models. Bhanu Prakash Doppala et.al [40] proposed a model that was evaluated on diverse datasets to determine its efficacy in improving accuracy. The evaluation involved testing the model on three datasets: the Cleveland dataset, a comprehensive dataset from IEEE Dataport, and a cardiovascular disease dataset from the Mendeley Data Center. The results of the proposed model exhibited high accuracy rates of 96.75%, 93.39%, and 88.24% on the respective datasets.

In contrast to the above work, the ensemble classifier is implemented using six ML models on the Cleveland heart disease dataset [41] and the IEEE Dataport heart disease datasets (comprehensive) [42]. This study used six ML algorithms: RF, KNN, LR, NB, GB, and AB. A GridsearchCV hyperparameter method and five-fold cross-validation methods were employed to obtain the best accuracy results before implementing the models. The hyperparameter values provided by GridsearchCV enhance the accuracy of the model. Using these parameters, the accuracy of six different algorithms is verified and the most accurate algorithm is determined. Additionally, the ensemble method was applied to the proposed algorithms in order to enhance their accuracy. This method boosts overall model accuracy from 90.16% (LR) to 93.44%, and from 90% (AB) to 95% using soft-voting ensemble classifiers on Cleveland and IEEE datasets.

## 3. Resources and Approaches

This section describes the methods to predict heart disease using the two benchmark publicly available datasets. This study consists of various phases, from the collection of data to the prediction of heart disease. In the first phase, data can be pre-processed using feature scaling and data transformation methods. The proposed model is built using multiple ML algorithms as the next step. An ensemble approach is used in the next phase of the process to enhance the model's accuracy. Figure 1 shows a detailed diagram of the workflow architecture.

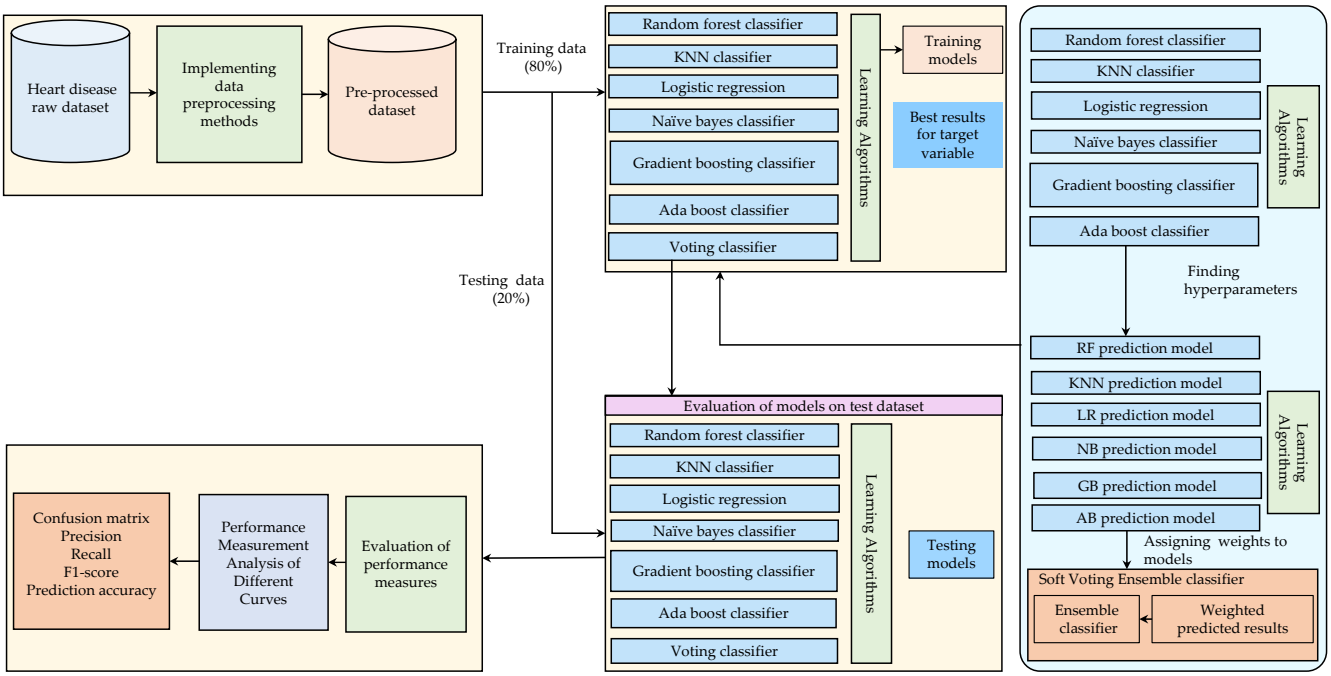


**Figure 1.** The proposed system model for predicting heart disease.

### *3.1. Description of the Datasets*

Dataset I: The UC Irvine ML Repository-Cleveland dataset, which contains 303 instances and 14 attributes, is included in this dataset.

Dataset II: The IEEE Dataport heart disease dataset (comprehensive) comprises 12 multivariate attributes and 1190 instances, which are included in this dataset.

The availability of these datasets can accelerate research in the field of heart disease prediction and lead to the development of more accurate and effective diagnostic tools.

### *3.2. Data Pre-Processing*

This study preprocesses Dataset I and Dataset II heart disease datasets before constructing the predictive model with ML. Since these datasets have undergone extensive preprocessing and cleaning, they are easier to use and require less time and effort for data preparation. Additionally, they are well documented and frequently cited in scientific literature.

In both datasets, a target attribute integer value indicates the presence of a patient's heart disease. If the value is 0, there is no heart disease, and if it is 1, there is heart disease. Based on gender, the attribute 'sex' consists of two classes: 1 for males and 0 for females. There are four classes of chest pain in the attribute 'cp' (chest pain type) two classes of fasting blood sugar in the attribute 'fbs' (fasting blood sugar) three classes of resting electrocardiograms in the attribute 'restecg' (resting ecg) and two classes of exercise in the attribute 'exang' (exercise angina). Additionally, 'slope' (ST slope) is composed of three classes. The remaining four attributes, such as 'trestbps' (resting bp s), 'chol' (cholesterol), 'age', and 'oldpeak', are considered numerical values. There are different steps involved in data pre-processing, from reading data to splitting data for training and testing. The steps are illustrated in Figure 2.

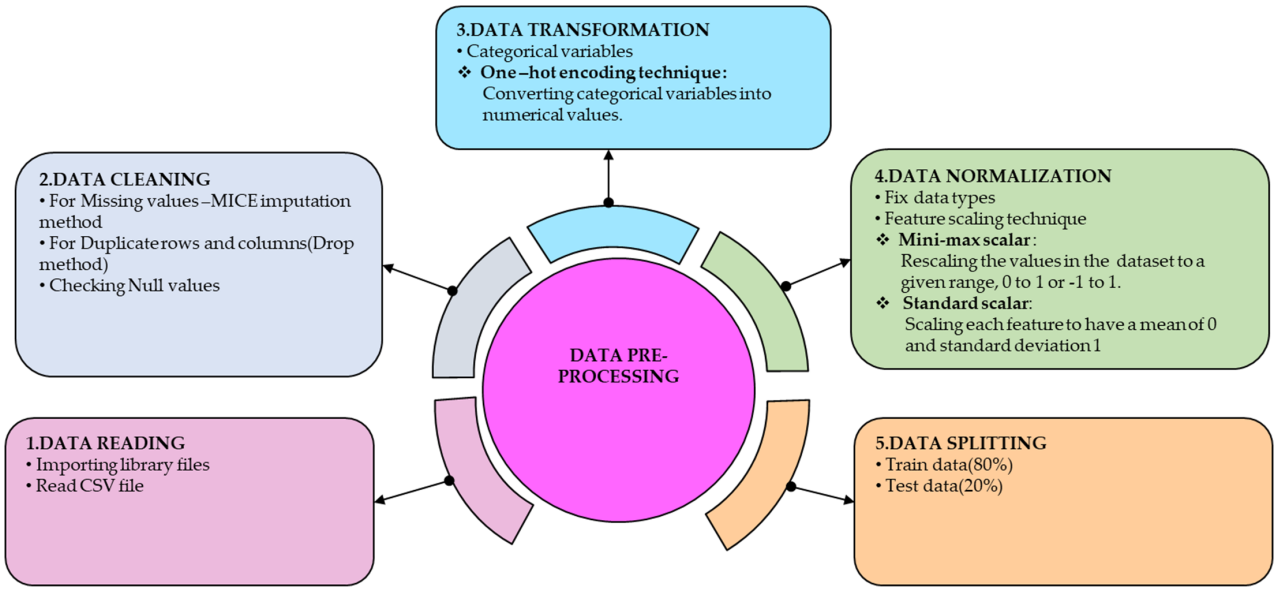


**Figure 2.** Data pre-processing stepwise diagram.

The data pre-processing process begins by identifying and addressing missing or duplicate values in the dataset. The presence of missing data in a dataset implies incompleteness, which can affect the statistical significance of the results obtained from the model analysis. When data are missing, the overall accuracy and validity of the analysis may be compromised. Therefore, to maximize the effectiveness of the analysis, it is recommended to fill in missing values either with a user-defined constant or the average value of the dataset, rather than completely suppressing the observations. The detailed explanation of each attribute presented in Tables 1 and 2 describes both datasets. Initially, Dataset I contained 303 instances, out of which one duplicate row was removed. As a result, a

dataset of 302 unique instances was obtained, with 164 instances corresponding to patients with heart disease and 138 instances corresponding to patients without heart disease. In Dataset II, we identified no missing values. Additionally, 272 duplicate instances have been identified. Therefore, to complete the dataset, the duplicate instance was removed. Among these, 508 instances correspond to patients with heart disease, and the remaining 410 instances belong to patients without heart disease.

**Table 1.** Attributes for the heart disease dataset.

| Sr. No. | Attribute Icon | Attribute Name | Description |
|---|---|---|---|
| 1 | Age | Age | Patient age |
| 2 | Sex | Gender | For males indicates 1, females 0 |
| 3 | Chest pain type | Chest pain type | Chest pain: (1) angina—0, (2) atypical angina—1, (3) non-anginal pain—2, and (4) asymptomatic—3. |
| 4 | Resting blood pressure | Rest state blood pressure (mm/Hg) | Resting blood pressure upon hospital admission, measured in mm/Hg. |
| 5 | Serum cholesterol | Serum cholesterol (fat) | Blood cholesterol level measured in mg/dL. |
| 6 | Fasting blood sugar | Fasting blood sugar (not eating) | If the blood sugar level is over 120 mg/dL after a fast of not eating overnight, it is considered to be high (1—true). If it is below 120 mg/dL, it is considered to be normal (0—false). |
| 7 | Resting ECG | Rest ECG test | An EsCG test result can be categorized as follows: 0 for a normal result, 1 for the presence of ST-T wave abnormality, and 2 for left ventricular hypertrophy. |
| 8 | Max. heart rate | Max. heart rate achieved | Max heart rate during exercise. |
| 9 | Exercise angina | Exercise-induced angina | Angina occurred by a workout, 0 for no; 1 for yes. |
| 10 | Oldpeak | ST depression (ECG test) | ST depression due to exercise relative to relaxation will observe in the ECG test |
| 11 | ST slope | Slope (ST depression) | Maximum workout 1 for upsloping; 2 for flat; 3 for down sloping) |
| 12 | Ca | No. of vessels (0–3) | The number of major blood vessels that can be visualized using fluoroscopy can range from 0 to 3. |
| 13 | Thal | Thalassemia (hemolytic disease) | Thalassemia is a blood disorder caused by abnormal hemoglobin production, with a score of 3 indicating normal production, 6 indicating permanent deficiency, and 7 signifying temporary impairment. |
| 14 | Target | Heart failure class attribute | No heart disease—0; heart disease—1 |

**Table 2.** Description of Datasets I, II.

| Datasets | Classes | Attributes | Instances |
|---|---|---|---|
| Cleveland (Dataset I) | 0—(no heart disease)<br>1—(heart disease) | 14 | 303 |
| IEEE Dataport (Dataset II) | 0—(no heart disease)<br>1—(heart disease) | 12 | 1190 |

However, some attributes in the data have large input values that are incompatible with other attributes, which results in poor learning performance. Therefore, to make it compatible with other attributes, data exploration was performed to visually explore and identify relationships between them. This is accomplished through the use of a one-hot encoding method. One-hot encoding is performed using features such as cp, thal, and slope for the available datasets. Those three features are further subdivided into cp_0 to cp_3,thal_0 to thal_3, and slope_0 to slope_2 features and merged into the original datasets. After exploring the data, the data were scaled for further processing. This is essential when using the dataset for a KNN. In order to make it compatible with all algorithms, a large number of features have been scaled down. As a result, ML models perform better.

Feature scaling involves two essential techniques called standardization and normalization. In standardization, the mean is subtracted from the distribution shifts and divided by the standard deviation. The act of subtracting the average from the data points is referred to as centering while dividing each data point by the standard deviation is called scaling. Standardization helps maintain the presence of outliers, making the resulting algorithm less susceptible to influences compared to one that has not undergone standardization. Standardizing a value can be accomplished using the following equations from (1) to (3).

$$x' = \frac{x - \mu}{\sigma} \tag{1}$$

Here, $x$ is the participation value, $x'$ is the standardized value, $\mu$ the mean, and $\sigma$ is the standard deviation. These can be calculated as follows:

$$\mu = \frac{\sum_{i=1}^{N} x}{N} \tag{2}$$

When referring to a dataset, N represents the total number of columns in the attribute being scaled. From the available dataset, age, trestbps, chol, and oldpeak features have large dimensional values. Hence, the standard scalar is used to convert these feature values into uniform scaling.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{3}$$

After scaling the large feature values, min-max scaling is applied for normalization. This technique is appropriate for data distributions that do not follow a Gaussian distribution. As a result of normalization, feature values become bounded intervals between the minimum and maximum. For min-max scaling, normalize the data using Equation (4) below.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{4}$$

Here, $x_{min}$ and $x_{max}$ are the minimum and maximum values of the respective feature in the dataset. With the use of the above equation, all the features are normalized [0,1]. The last step in pre-processing involved dividing the data into two subsets, known as training and testing data, after normalizing the data. The split was carried out in such a way that 80% of the available data was allocated for training and the remaining 20% for testing. This division enabled the training and evaluation of various ML classifiers by testing their accuracy using the training and testing datasets. An exploratory data analysis (EDA) is also conducted prior to discussing each algorithm used to predict heart disease. A description of the descriptive statistics and the information regarding the correlation matrix cannot be presented here for brevity.

### *3.3. Performance Measures*

This study applied various ML algorithms such as RF, KNN, LR, NB, GB, and AB to predict heart disease. Before using the ML algorithms, a number of matrices such as the confusion matrix, receiver operating characteristics (ROC), the area under curve (AUC), learning curve, and precision-recall curve are briefly described in the following subsection.

1. Confusion matrix

The confusion matrix provides a visual representation of the algorithm's performance. The confusion matrix table makes it easy to visually inspect the prediction errors. The confusion matrix depicted in Figure 3 comprises four components: true negatives (TNs), false positives (FPs), false negatives (FNs), and true positives (TPs). The matrix showcases actual class instances as rows and predicted class instances as columns (or vice versa) [43]. The confusion matrix serves not only as a visual representation of errors, but can also

include various metrics such as precision, recall, and F1. Each metric holds its significance and is applied in specific situations.

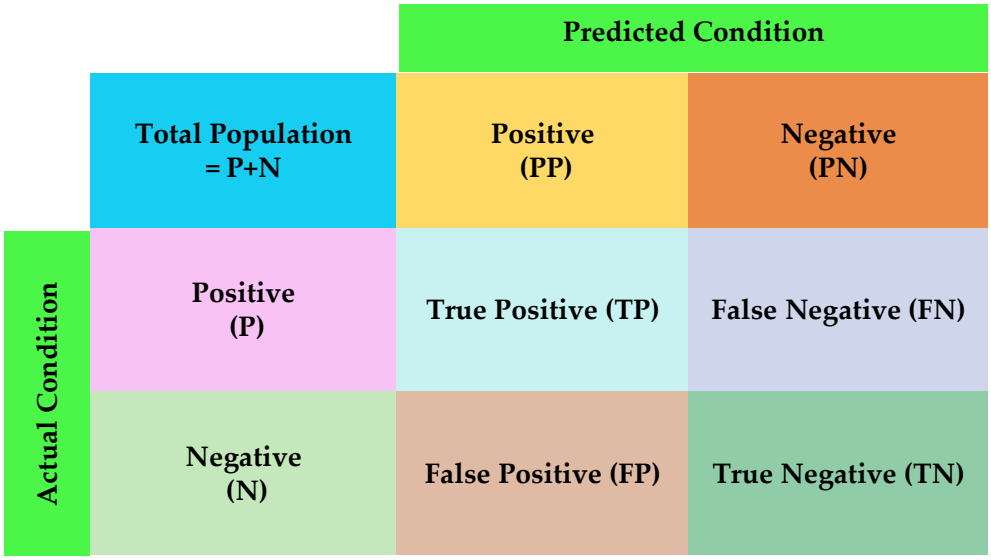


**Figure 3.** The confusion matrix.

- Precision

It is calculated based on the total number of predictions made by the model. The percentage of correct predictions is then divided by the total number of predictions [44]. This can be defined as the ratio of the TP to the total prediction (TP + FP) made by the model. It can be expressed as an equation in (5).

$$\text{Pr}\, ecision = \frac{TP}{TP + FP} \tag{5}$$

- Precision

A second significant metric is recall, which is also known as sensitivity or the true positive rate [45]. This can be determined by determining the proportion of positive observations that were accurately predicted in relation to the overall number of positive observations. Thus, recall indicates the range of positive classes. As an equation, it can be written as (6).

$$\text{Re}\, call = \frac{TP}{TP + FN} \tag{6}$$

A good classifier should have precision and recall of one, which corresponds to a FP and FN equal to zero. It is better to consider both precision and recall if the cost of the FP and FN is very different. Consequently, precision and recall need to be considered when there is an uneven distribution of classes. Therefore, the F1 score can be regarded as a measure of both precision and recall [46].

- Precision

The F1 score is obtained by taking the average of precision and recall. This metric has generally been considered to be a reliable method for comparing the performance of different classifiers, particularly when the data are unbalanced. F1 scores are calculated by considering both the number of prediction errors and the type of errors the model makes. As an equation, it can be written as (7).

$$F1\text{-}Score = \frac{2 \times (\text{Pr}\, ecision \times recall)}{\text{Pr}\, ecision + \text{Re}\, call} \tag{7}$$

2. ROC curve and AUC

ROC curves are utilized as a means of evaluating the performance of classification algorithms. The curve plots the true positive rate (TPR), also referred to as recall, against the false positive rate (FPR) at various threshold values [47]. The TPR is calculated using Equation (6), while the FPR is determined through Equation (8). This representation helps to distinguish between the actual positive results and false results (noise).

$$FPR = \frac{FP}{FP + TN} \tag{8}$$

The TPR is plotted on the $Y$-axis, while FPR is plotted on the $X$-axis. Thus, it is necessary to utilize a method referred to as AUC in order to calculate the values at any threshold level efficiently [48]. AUC measures the performance of a classifier across different thresholds as indicated by the ROC curve. In general, the AUC value ranges from 0 to 1, which suggests a good model will have an AUC close to 1, which indicates a high degree of separation. The ROC curve represents how well a classification model performs across all classification thresholds. On this curve, two parameters are plotted. The ROC space is divided by the diagonal. Points above the diagonal indicate successful classification; points below the line indicate unsuccessful classification. The valuation of the AUC curve is explained in Table 3.

**Table 3.** Valuation of the area under the curve.

| Area Under the Curve (AUC) | Understanding |
|---|---|
| $0.90 \leq$ AUC | Exceptional |
| $0.80 \leq$ AUC $< 0.90$ | Decent |
| $0.70 \leq$ AUC $< 0.80$ | Reasonable |
| $0.60 \leq$ AUC $< 0.70$ | Unfortunate |
| $0.50 \leq$ AUC $< 0.60$ | Flop |

3. ROC curve and AUC

Using a learning curve, we can determine how much more training data will benefit our model. It illustrates the relationship between training and test scores for a ML model with a variable number of training samples. The cross-validation procedure is carried out behind the scenes when we call the learning curve.

4. ROC curve and AUC

Plotting recall on the $x$-axis and precision on the $y$-axis obtains the precision-recall curve. This curve depicts the false positive to false negative ratio. The precision-recall curve is not constructed using the number of true negative results [49].

### *3.4. Accuracy and Loss of Each Fold Measurement*

In ML classifiers, the accuracy and loss of each fold have a significant impact on the model's overall performance. The accuracy of each fold determines how well the model has learned from the training data and how accurately it can predict new data. If the accuracy of a fold is high, it indicates that the model has successfully learned the underlying patterns in the data and can make accurate predictions. However, if the accuracy of a fold is low, it implies that the model needs further improvement and fine-tuning to achieve better results.

Similarly, the loss function of each fold plays a crucial role in determining the model's performance. The loss function measures how well the model can approximate the actual values of the target variable. A low loss value indicates that the model is fitting the training data well and has the potential to perform well on new data. On the other hand, a high loss value suggests that the model is not fitting the training data well, and more refinement is necessary to improve its performance. Finally, both accuracy and loss of each fold are essential metrics that impact the performance of ML classifiers.

- Log loss function

Log loss, also known as cross-entropy loss, is a measure of the performance of a classification model. It measures the difference between the predicted probabilities of the model and the actual outcomes. In binary classification problems, the log loss formula can be expressed as in Equation (9).

$$\log\_loss = -((xy)\log(p) + x(1-y)\log(1-p)) \quad (9)$$

where y is the true label (either 0 or 1), p is the predicted probability of the positive class, and the log is the natural logarithm. The log loss ranges from 0 to infinity, with a perfect model having a log loss of 0. A model that always predicts the same probability for all samples would have a log loss of approximately 0.693. Log loss penalizes highly confident but wrong predictions more than it penalizes predictions that are only slightly wrong. As a result, it is a popular loss function for classification problems where the focus is on predicting probabilities rather than hard class labels.

## 4. ML Classification Algorithms and Experimental Data Analysis

### *4.1. Hyperparameter Tuning and Experimental Results*

Optimizing an ML model's performance is essential before its implementation to ensure it achieves the highest possible precision. This optimization process entails the careful adjustment of specific variables called hyperparameters, which govern the model's learning behavior. Fine-tuning a model typically involves fitting it to a training dataset multiple times with various hyperparameter combinations, ultimately determining the ideal configuration for improved performance.

One efficient method for exploring the optimal hyperparameter values is with GridSearchCV, a technique that involves creating a comprehensive grid of potential hyperparameter values. Tables 4 and 5 provide a list of hypermeter tuning values for six ML classifiers.

**Table 4.** Hypermeter tuning values for Dataset I.

| Sr. No. | Classifier | GridsearchCV Hypermeter Tuning Values |
|---|---|---|
| 1 | RF | n_estimators = 500, random_state = 42, max_leaf_nodes = 20, min_samples_split = 15 |
| 2 | KNN | n_neighbors = 19 |
| 3 | LR | max_iter = 20, random_state = 1, solver = 'newton-cg', penalty = l2 |
| 4 | NB | var_smoothing = 0.35 |
| 5 | GB | learning_rate = 0.2, n_estimators = 50, max_depth = 3 |
| 6 | AD | n_estimators = 10, learning_rate = 0.6 |

**Table 5.** Hypermeter tuning values for Dataset II.

| Sr. No. | Classifier | GridsearchCV Hypermeter Tuning Values |
|---|---|---|
| 1 | RF | 'criterion': 'gini', 'max_depth': 8, 'max_features': 'auto', 'n_estimators': 200 |
| 2 | KNN | 'n_neighbors': 12 |
| 3 | LR | class_weight = 'balanced', max_iter = 20, random_state = 42, solver = 'liblinear' |
| 4 | NB | 'var_smoothing': 0.8111308307896871 |
| 5 | GB | 'learning_rate': 0.2, 'max_depth': 3, 'n_estimators': 50 |
| 6 | AD | 'learning_rate': 0.6, 'n_estimators': 10 |

### *4.2. Random Forest Classifier*

The RF classifier makes predictions by averaging the predictions of their real trees. RF is an ensemble-learning-based method for supervised ML [50]. It utilizes bagging to combine multiple decision trees, thereby improving the accuracy of predictions. Bagging training is provided on an individual basis to each individual. As part of the training process, each decision tree is evaluated using different samples of data that were generated randomly using replacements from the original dataset. When constructing trees, a random

selection of features is also made. A majority vote is used to combine the predictions of multiple trees [51]. For Dataset I, the model's confusion matrix revealed that it successfully predicted 19 positive cases and 33 negative cases. However, there were nine incorrect predictions, consisting of eight false negatives and one false positive. In the case of Dataset II, the confusion matrix showed that the model accurately predicted 71 positive cases and 92 negative cases, but it also made 21 incorrect predictions, which included 17 false negatives and 4 false positives. Table 6 showcases the performance of the RF in predicting heart disease for two datasets: Dataset I (Cleveland) and Dataset II (IEEE Dataport). The metrics used to evaluate the model include precision, recall, and F1 score, for both classes, 0 (no heart disease) and 1 (having heart disease).

**Table 6.** Performance measure curve values of RF (Datasets I and II).

| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
|---|---|---|---|---|---|---|---|
| | | Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
| | 0 | 95 | 70 | 81 | 94 | 82 | 87 |
| | 1 | 80 | 97 | 88 | 85 | 95 | 90 |
| Accuracy | | | | 85 | | | 89 |
| Macro average | | 88 | 84 | 84 | 89 | 88 | 88 |
| Weighted average | | 87 | 85 | 85 | 89 | 89 | 89 |

For Dataset I, Class 0 has a precision of 95%, recall of 70%, F1 score of 81%, and 27 instances. Class 1 has a precision of 80%, recall of 97%, F1 score of 88%, and 34 instances. The overall accuracy, macro average, and weighted average are 85%, 88%, and 87%, respectively, for the 61-instance dataset. For Dataset II, Class 0 has a precision of 94%, recall of 82%, F1 score of 87%, and 88 instances. Class 1 has a precision of 85%, recall of 95%, F1 score of 90%, and 96 instances. The overall accuracy, macro average, and weighted average are 89% for the 184-instance dataset. Figures 4 and 5 represent the RF model's performance measuring plots on Dataset I and Dataset II.

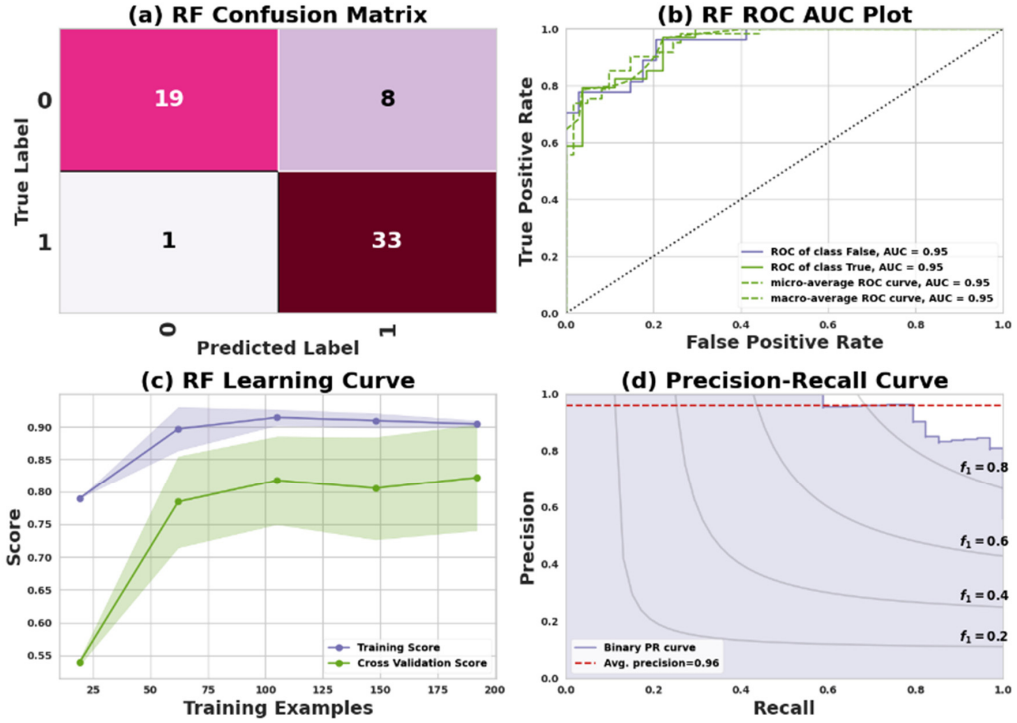


**Figure 4.** Performance measuring curves of RF on Dataset I.

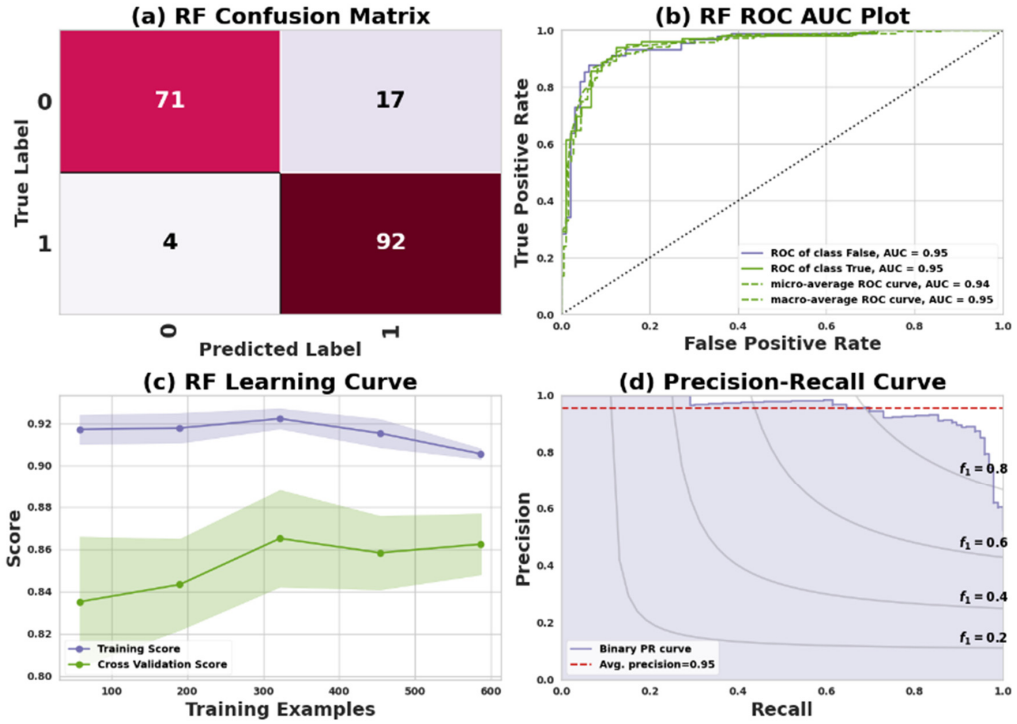


**Figure 5.** Performance measuring curves of RF on Dataset II.

### *4.3. K-Nearest Neighbor Classifier*

KNN is an instance-based or lazy learning technique. The term lazy learning refers to the process of building a model without the requirement of training data. KNN neighbors are selected from a set of objects with known properties or classes [52]. The confusion matrix reveals that for Dataset I, 22 positive records and 29 negative records were accurately predicted, while 10 predictions were inaccurate, specifically consisting of 5 false negatives and 5 false positives. Similarly, in the confusion matrix for Dataset II, 74 positive records and 88 negative records were correctly predicted, but 22 predictions were inaccurate, including 14 false negatives and 8 false positives. Table 7 presents the performance of the KNN classifier in predicting heart disease for two datasets: Dataset I (Cleveland) and Dataset II (IEEE Dataport). Evaluation metrics include precision, recall, F1 score, and support for both classes: 0 (no heart disease) and 1 (having heart disease).

**Table 7.** Performance measure curve values of KNN (Datasets I and II).

| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
|---|---|---|---|---|---|---|---|
| | | Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
| | 0 | 81 | 81 | 81 | 90 | 84 | 87 |
| | 1 | 85 | 85 | 85 | 86 | 92 | 89 |
| Accuracy | | | | 84 | | | 88 |
| Macro average | | 83 | 83 | 83 | 88 | 88 | 88 |
| Weighted average | | 84 | 84 | 84 | 89 | 88 | 88 |

In Dataset I, Class 0 shows a precision of 81%, recall of 81%, F1 score of 81%, and 27 instances. Class 1 displays a precision of 85%, recall of 85%, F1 score of 85%, and 34 instances. The dataset, containing 61 instances, has an overall accuracy, macro average, and weighted average of 84%, 83%, and 84%, respectively. For Dataset II, Class 0 has a precision of 90%, recall of 84%, F1 score of 87%, and 88 instances. Class 1 demonstrates a precision of 86%, recall of 92%, F1 score of 89%, and 96 instances. The overall accuracy, macro average, and weighted average for the 184-instance dataset are 88%, 88%, and 89%,

respectively. Figures 6 and 7 represent the KNN model's performance measuring plots on Dataset I and Dataset II.

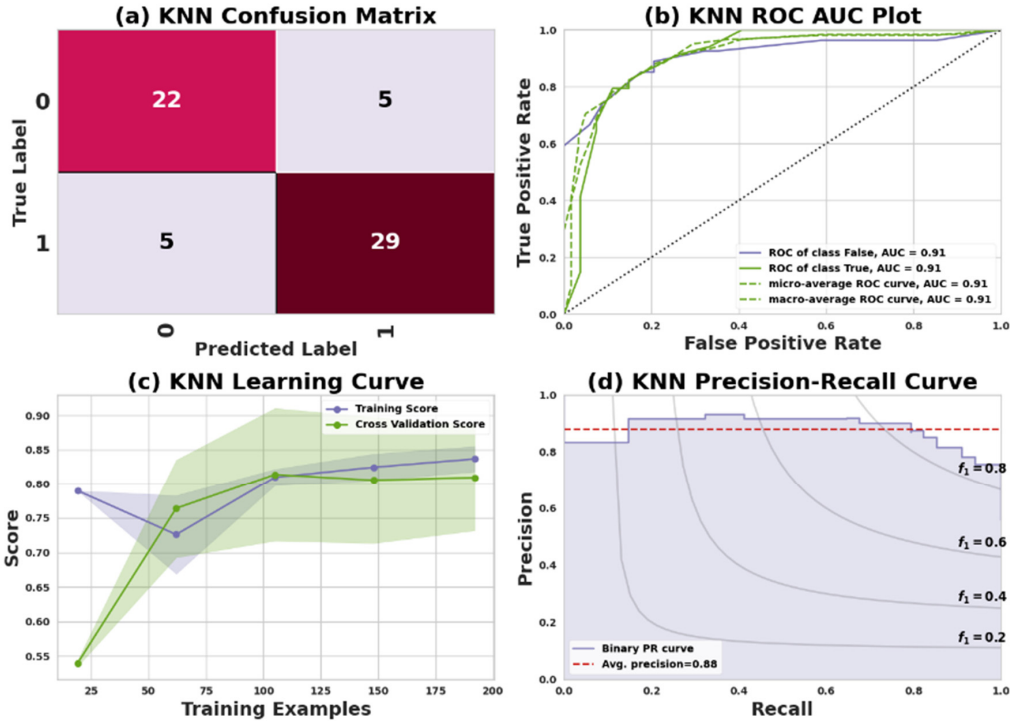


**Figure 6.** Performance measuring curves of KNN on Dataset I.

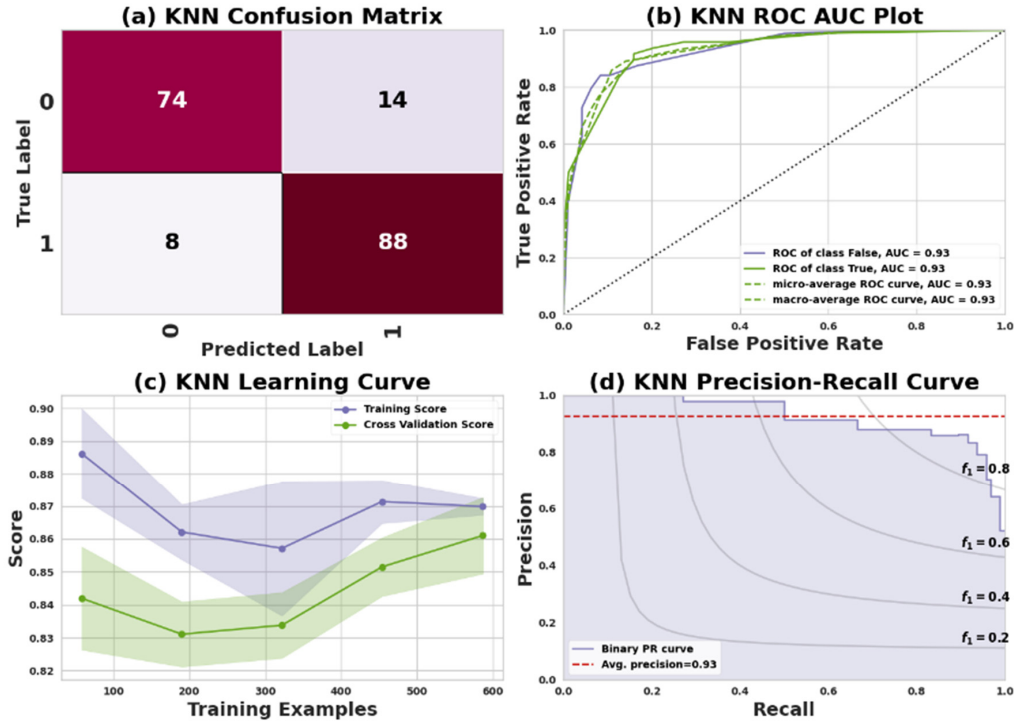


**Figure 7.** Performance measuring curves of KNN on Dataset II.

### *4.4. Logistic Regression Classifier*

LR is an algorithm for predicting whether an observation belongs to one of two categories in ML. The LR classifiers predict the target class based on calculated logits (scores). A logistic function is used to convert probabilities into binary values that can be used to

make predictions [53]. The confusion matrix for the model reveals the following results for Dataset I and Dataset II: In Dataset I, the model accurately predicted 21 positive and 34 negative cases while making 6 incorrect predictions, all of which were false negatives and no false positives. In Dataset II, the model successfully predicted 75 positive and 88 negative cases, but it also made 21 incorrect predictions, comprising 13 false negatives and 8 false positives. Table 8 illustrates the performance of a Logistic Regression (LR) classifier in predicting heart disease for two datasets: Dataset I (Cleveland) and Dataset II (IEEE Dataport). The evaluation metrics presented include precision, recall, F1 score, and support for both classes: 0 (no heart disease) and 1 (having heart disease).

**Table 8.** Performance measure curve values of LR (Datasets I and II).

| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
|---|---|---|---|---|---|---|---|
| | | Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
| | 0 | 100 | 78 | 88 | 90 | 85 | 88 |
| | 1 | 85 | 100 | 92 | 87 | 92 | 89 |
| Accuracy | | | | 90 | | | 89 |
| Macro average | | 93 | 89 | 90 | 89 | 89 | 89 |
| Weighted average | | 92 | 90 | 90 | 89 | 89 | 89 |

In Dataset I (Cleveland), Class 0 has a precision of 100%, recall of 78%, F1 score of 88%, and 27 instances. Class 1 exhibits a precision of 85%, recall of 100%, F1 score of 92%, and 34 instances. With a total of 61 instances, the overall accuracy, macro average, and weighted average are 90%, 93%, and 92%, respectively.

For Dataset II (IEEE Dataport), Class 0 displays a precision of 90%, recall of 85%, F1 score of 88%, and 88 instances. Class 1 shows a precision of 87%, recall of 92%, F1 score of 89%, and 96 instances. The dataset, containing 184 instances, has an overall accuracy, macro average, and weighted average of 89%. Figures 8 and 9 represent the LR model's performance measuring plots on Dataset I and Dataset II.

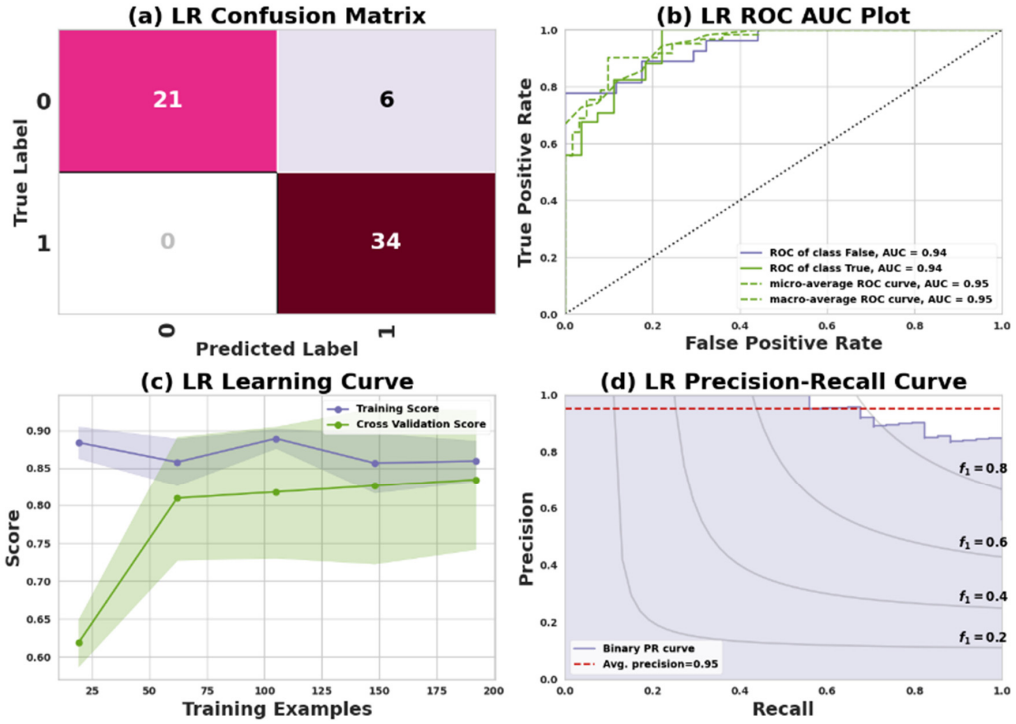


**Figure 8.** Performance measuring curves of LR on Dataset I.

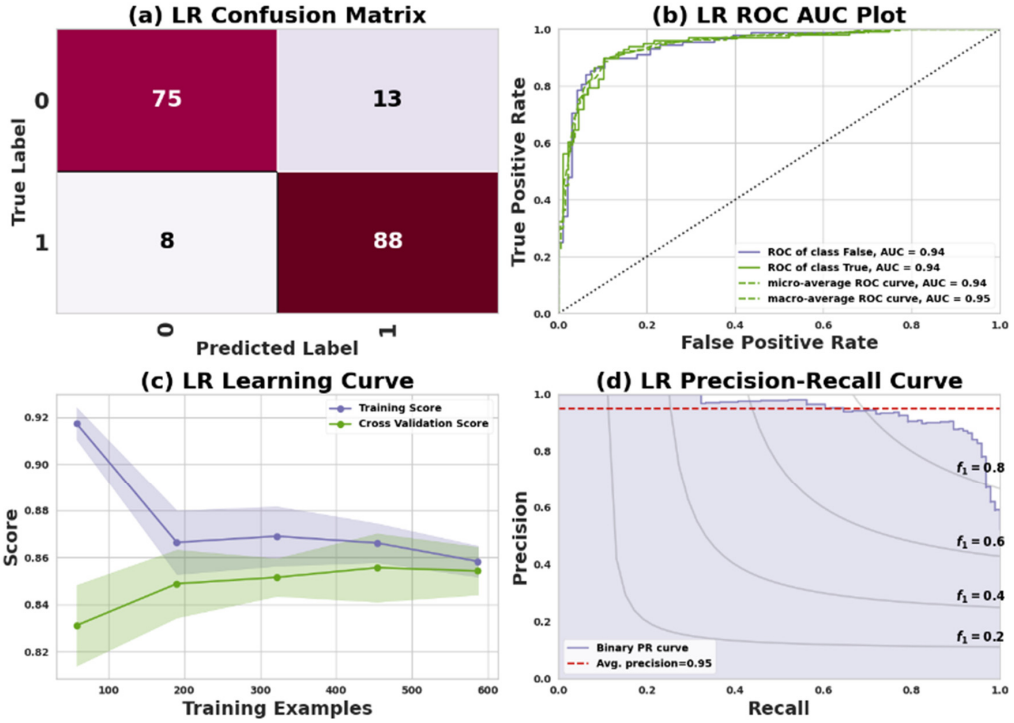


**Figure 9.** Performance measuring curves of LR on Dataset II.

### *4.5. Naive Bayes Classifier*

One of the most popular supervised ML algorithms for multi-classification problems is the NB algorithm. Several classification problems can be solved using the NB algorithm, which is based on the Bayes theorem. The basic concept of NB is to estimate the probability of each class we wish to reveal based on the probability of each feature being present in the data. According to Equation (10), naive models assume that the features of a model are independent of each other.

$$P(c/x) = \frac{p(x/c)p(c)}{p(x)} \tag{10}$$

where the P(c/x) represents the posterior probability, which is the probability of a hypothesis (or class) given the observed data. The term p(x/c) denotes the likelihood, which is the probability of observing the data given the hypothesis (or class). The class prior to probability is denoted by p©, and it represents the probability of observing the hypothesis (or class) in the absence of any data. Finally, the predictor prior to probability, denoted by p(x), represents the probability of observing the data in the absence of any hypothesis (or class) [54]. The NB algorithm assumes that each feature in the data has an independent condition on how the probability of an outcome will happen for each unique class of data in the dataset. For Dataset I, the confusion matrix reveals that the model accurately predicted 23 positive and 31 negative cases, while making 7 incorrect predictions, which include 4 false negatives and 3 false positives. In the case of Dataset II, the confusion matrix shows that the model successfully predicted 73 positive and 89 negative cases, but also made 22 incorrect predictions, comprising 15 false negatives and 7 false positives. Table 9 presents the performance of a NB model in predicting heart disease for two datasets: Dataset I (Cleveland) and Dataset II (IEEE Dataport). Evaluation metrics include precision, recall, F1 score, and support for both classes: 0 (no heart disease) and 1 (having heart disease).

**Table 9.** Performance measure curve values of NB (Datasets I and II).

| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
|---|---|---|---|---|---|---|---|
| | | Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
| | 0 | 88 | 85 | 87 | 88 | 85 | 87 |
| | 1 | 89 | 91 | 90 | 89 | 91 | 90 |
| Accuracy | | | | 89 | | | 89 |
| Macro average | | 89 | 88 | 88 | 89 | 88 | 88 |
| Weighted average | | 89 | 89 | 88 | 89 | 89 | 88 |

In Dataset I, Class 0 has a precision of 88%, recall of 85%, F1 score of 87%, and 27 instances. Class 1 exhibits a precision of 89%, recall of 91%, F1 score of 90%, and 34 instances. With 61 instances in total, the overall accuracy, macro average, and weighted average are 89%.

For Dataset II, Class 0 displays a precision of 88%, recall of 85%, F1 score of 87%, and 88 instances. Class 1 shows a precision of 89%, recall of 91%, F1 score of 90%, and 96 instances. With 184 instances, the overall accuracy, macro average, and weighted average are 89%. Figures 10 and 11 represent the NB model's performance measuring plots on Dataset I and Dataset II.

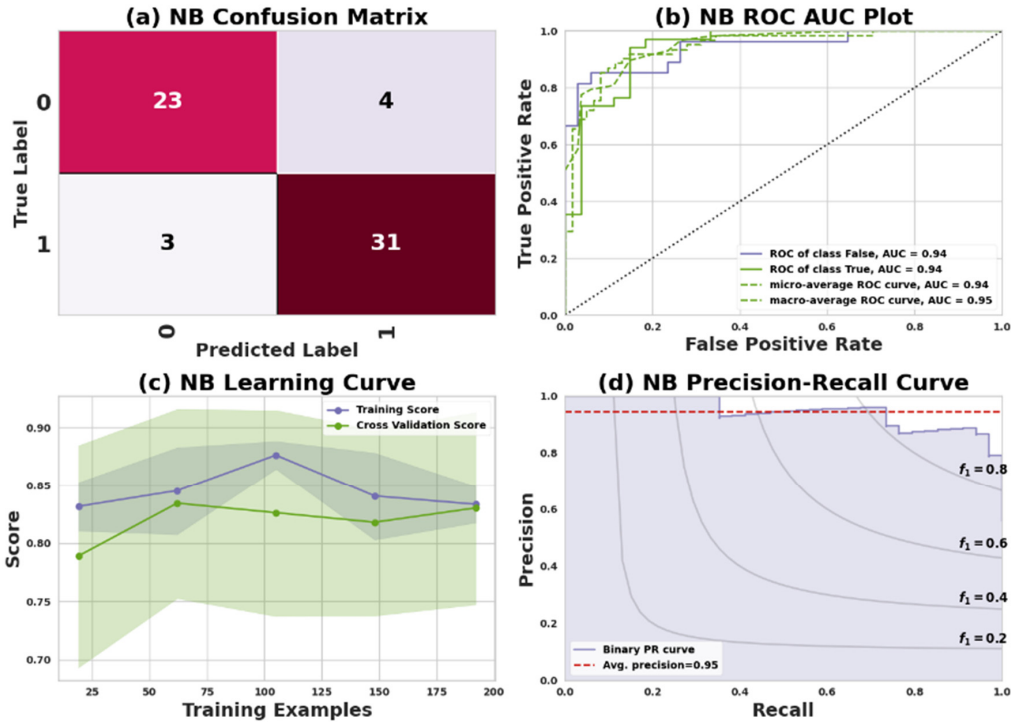


**Figure 10.** Performance measuring curves of NB on Dataset I.

### *4.6. Gradient Boosting Classifier*

A GB Classifier is an ML technique that uses an ensemble of weak models to produce a robust classifier. The algorithm sequentially trains individual models, each addressing the residual errors generated by the previous model. The final prediction is made by combining each model's predictions, weighted according to their contribution [55]. This technique can be applied to binary and multi-class classification problems and is often implemented using decision trees for weak learners. The goal is to minimize the loss function through the iterative process of model training and combining. For Dataset I, the confusion matrix indicates that the model accurately predicted 21 positive and 31 negative cases, while making 9 incorrect predictions, consisting of 6 false negatives and 3 false positives. In Dataset II, the confusion matrix reveals that the model successfully predicted 75 positive and 89 negative cases, but also made 20 incorrect predictions, which include 13 false

negatives and 7 false positives. Table 10 showcases the performance of a GB classifier in predicting heart disease for two datasets: Dataset I (Cleveland) and Dataset II (IEEE Dataport). It includes evaluation metrics such as precision, recall, F1 score, and support for both classes: 0 (no heart disease) and 1 (having heart disease).

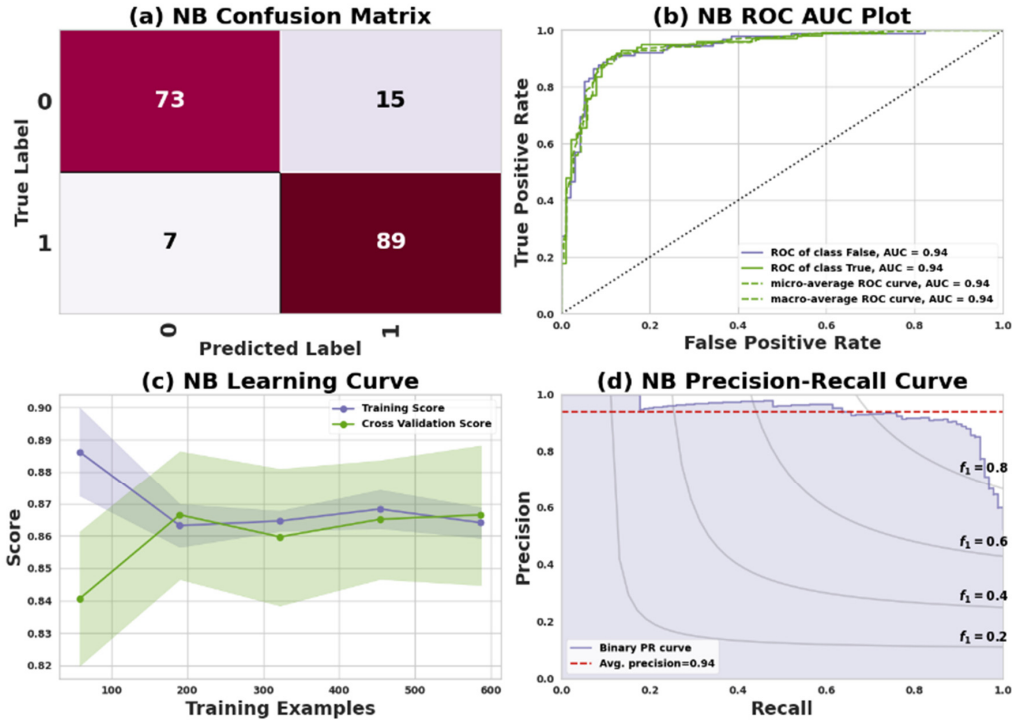


**Figure 11.** Performance measuring curves of NB on Dataset II.

**Table 10.** Performance measure curve values of GB (Datasets I and II).

| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
|---|---|---|---|---|---|---|---|
| | | Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
| | 0 | 88 | 78 | 82 | 91 | 85 | 88 |
| | 1 | 84 | 91 | 87 | 87 | 93 | 90 |
| Accuracy | | | | 85 | | | 89 |
| Macro average | | 86 | 84 | 85 | 89 | 89 | 89 |
| Weighted average | | 85 | 85 | 85 | 89 | 89 | 89 |

For Dataset I (Cleveland), Class 0 has a precision of 88%, recall of 78%, F1 score of 82%, and 27 instances. Class 1 displays a precision of 84%, recall of 91%, F1 score of 87%, and 34 instances. The dataset, with 61 instances, has an overall accuracy, macro average, and weighted average of 85%.

In Dataset II (IEEE Dataport), Class 0 exhibits a precision of 91%, recall of 85%, F1 score of 88%, and 88 instances. Class 1 presents a precision of 87%, recall of 93%, F1 score of 90%, and 96 instances. With 184 instances, the overall accuracy, macro average, and weighted average are 89%. Figures 12 and 13 represent the GB model's performance measuring plots on Dataset I and Dataset II.

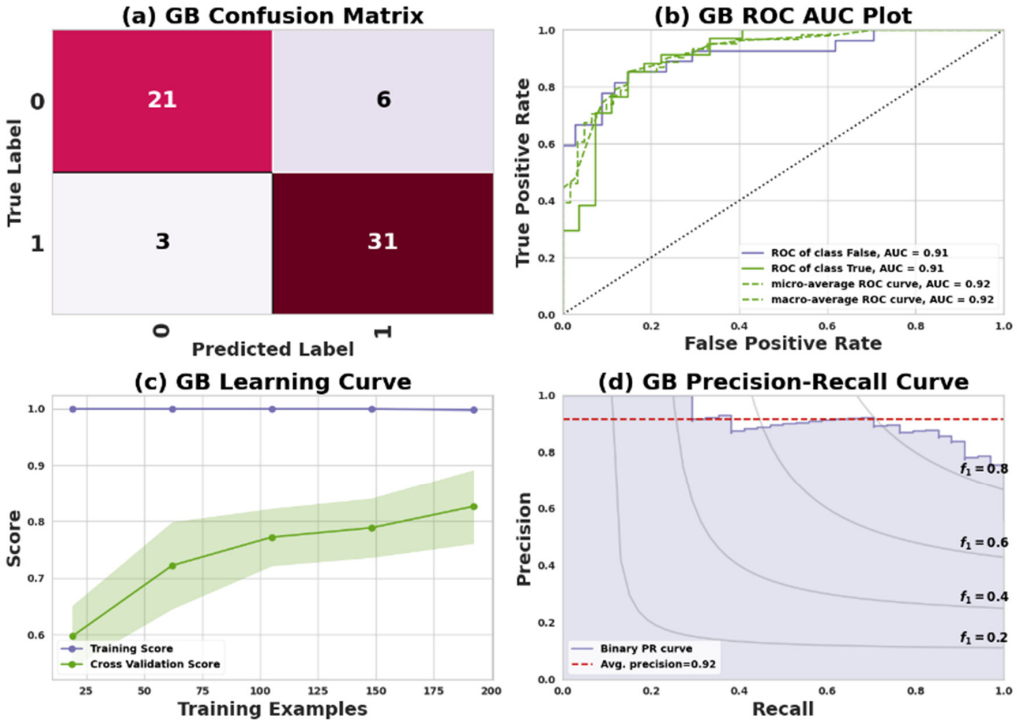


**Figure 12.** Performance measuring curves of GB on Dataset I.

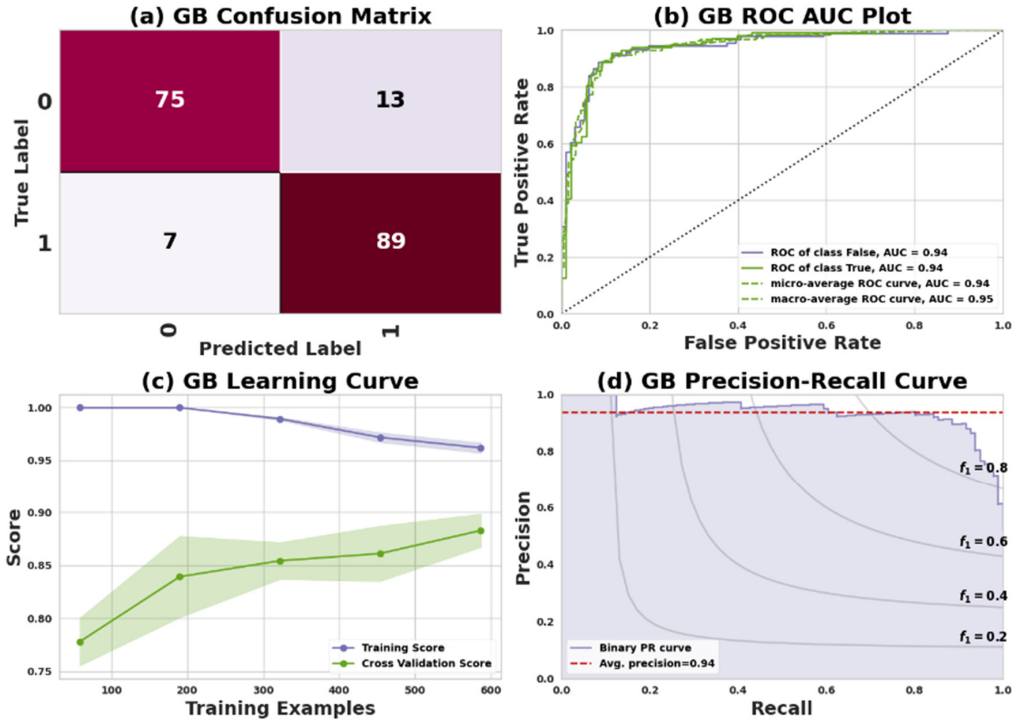


**Figure 13.** Performance measuring curves of GB on Dataset II.

### 4.7. AdaBoost Classifier

The AB is a ML algorithm designed for classification tasks. It combines multiple simple models, known as weak learners, to form a more robust overall classifier. The algorithm starts by training the first weak learner on the data and then calculates the error. Subsequently, misclassified samples are given greater weight, and the subsequent weak learner is trained on these samples with higher emphasis. This process is repeated several times. Each weak learner's prediction is given a weight proportional to its accuracy before being combined to form the final prediction [56]. The AB can be used for binary or multi-class classification problems, and weak learners often utilize decision trees. The algorithm adjusts the weight of the samples based on their classification performance, allowing it to focus on the samples that are challenging to classify. For Dataset I, the confusion matrix shows that the model accurately predicted 19 positive and 33 negative cases, while making 9 incorrect predictions, which include 8 false negatives and 1 false positive. In Dataset II, the confusion matrix indicates that the model successfully predicted 75 positive and 90 negative cases, but also made 19 incorrect predictions, consisting of 13 false negatives and 6 false positives. Table 11 presents the performance of the AB classifier in predicting heart disease for two datasets: Dataset I (Cleveland) and Dataset II (IEEE Dataport). Evaluation metrics include precision, recall, F1 score, and support for both classes: 0 (no heart disease) and 1 (having heart disease).

**Table 11.** Performance measure values of AB classifier (Dataset I and II).

| Model Accuracy | Classes | Dataset I (Cleveland) | | | Dataset II (IEEE Dataport) | | |
|---|---|---|---|---|---|---|---|
| | | Precision (%) | Recall (%) | F1 Score (%) | Precision (%) | Recall (%) | F1 Score (%) |
| | 0 | 95 | 70 | 81 | 93 | 85 | 89 |
| | 1 | 80 | 97 | 88 | 87 | 94 | 90 |
| Accuracy | | | | 85 | | | 90 |
| Macro average | | 88 | 84 | 84 | 90 | 89 | 90 |
| Weighted average | | 87 | 85 | 85 | 90 | 90 | 90 |

In Dataset I (Cleveland), Class 0 has a precision of 95%, recall of 70%, F1 score of 81%, and 27 instances. Class 1 demonstrates a precision of 80%, recall of 97%, F1 score of 88%, and 34 instances. With 61 instances in total, the overall accuracy, macro average, and weighted average are 85%.

For Dataset II (IEEE Dataport), Class 0 shows a precision of 93%, recall of 85%, F1 score of 89%, and 88 instances. Class 1 displays a precision of 87%, recall of 94%, F1 score of 90%, and 96 instances. With 184 instances, the overall accuracy, macro average, and weighted average are 90%. Figures 14 and 15 represent the AB model's performance measuring plots on Dataset I and Dataset II.

### 4.8. Performance Measurement Analysis of Different Curves

For each classifier, Table 12 shows the results obtained for both Dataset I and Dataset II. The performance is compared across different classifiers and datasets to understand which model performs better in each scenario. In general, the result indicates that the RF, LR, NB, and AB classifiers demonstrate higher ROC-AUC and precision-recall values, suggesting better overall performance compared to the other models. However, the GB classifier shows a perfect learning curve score of 100% for Dataset I, which implies it effectively learns from the training data. The ROC-AUC curve, learning curve, and precision-recall curves for both datasets are illustrated in Figures 4–15. This visualization allows for a comprehensive comparison of classifier performance across the two datasets, considering multiple evaluation metrics.

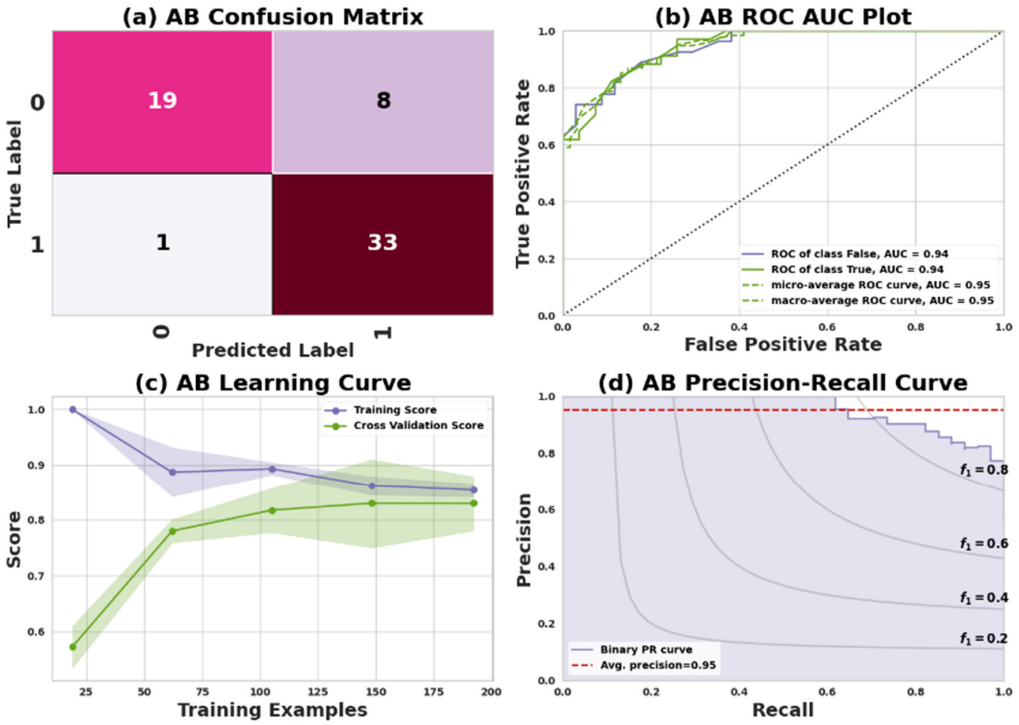


**Figure 14.** Performance measuring curves of AB on Dataset I.

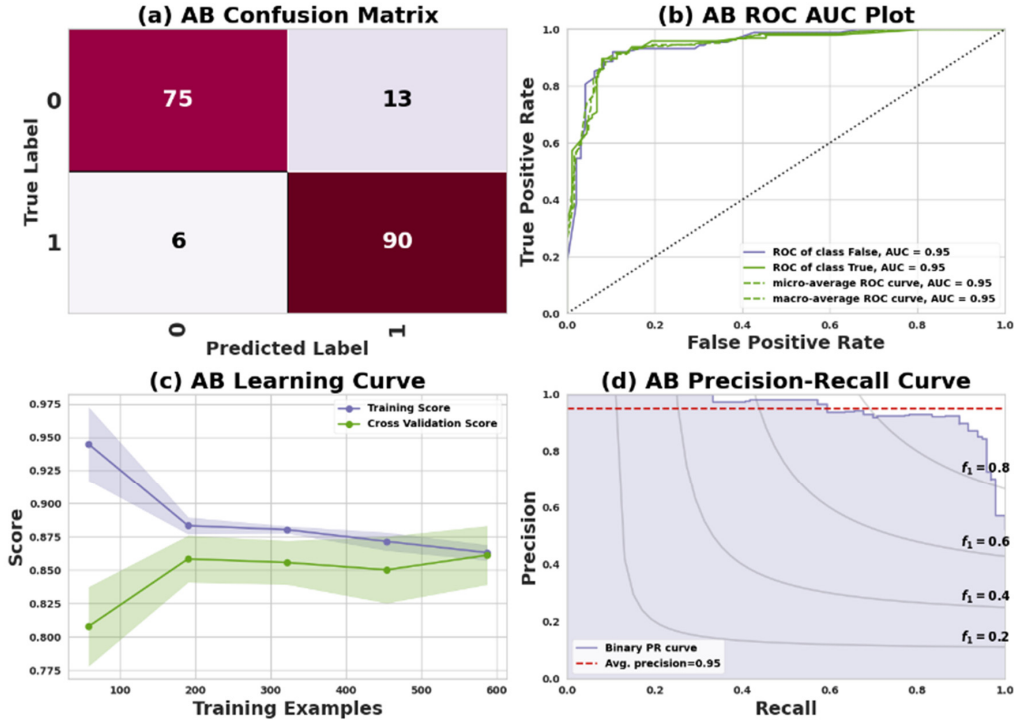


**Figure 15.** Performance measuring curves of AB on Dataset II.

### *4.9. Assessing the Accuracy and Accuracy Loss of Each Fold: Measurement and Performance Evaluation*

The loss and accuracy values for each fold provide an estimate of how well the model is performing on different subsets of the data. Figures 16–27 show six models' five-fold accuracy and loss value plots for Datasets I and II. Table 13 presents the values of accuracy, accuracy loss of each fold, and mean and standard deviation values of six models.

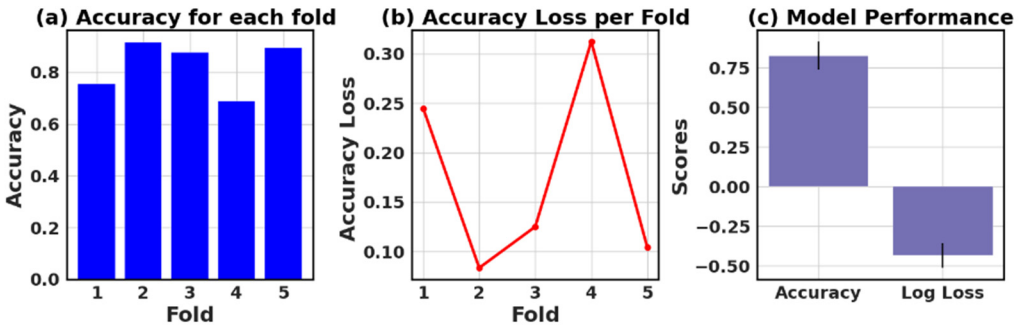


**Figure 16.** RF model's 5-fold accuracy and loss value plots for Dataset I.

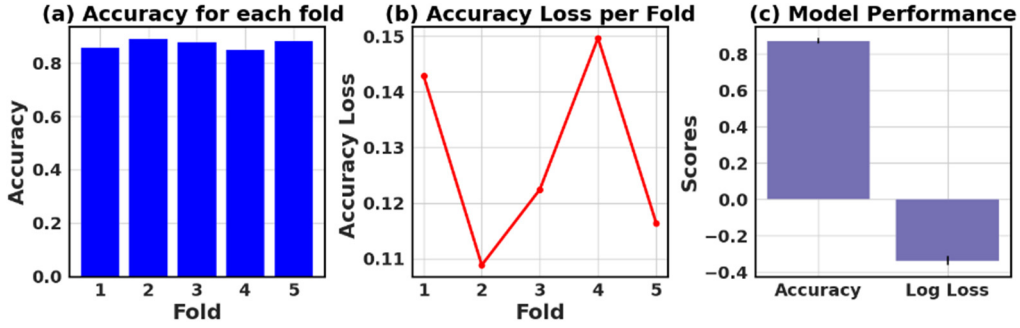


**Figure 17.** RF model's 5-fold accuracy and loss value plots for Dataset II.

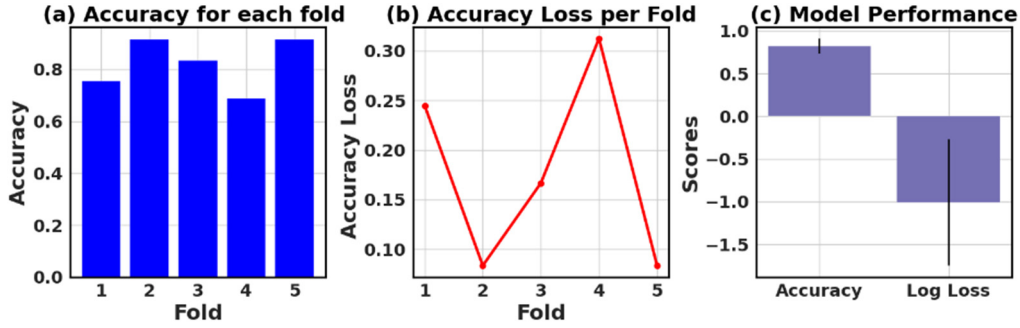


**Figure 18.** KNN model's 5-fold accuracy and loss value plots for Dataset I.

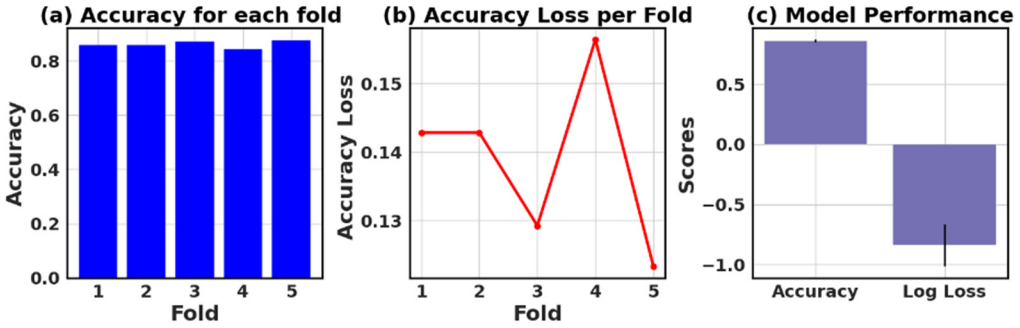


**Figure 19.** KNN model's 5-fold accuracy and loss value plots for Dataset II.

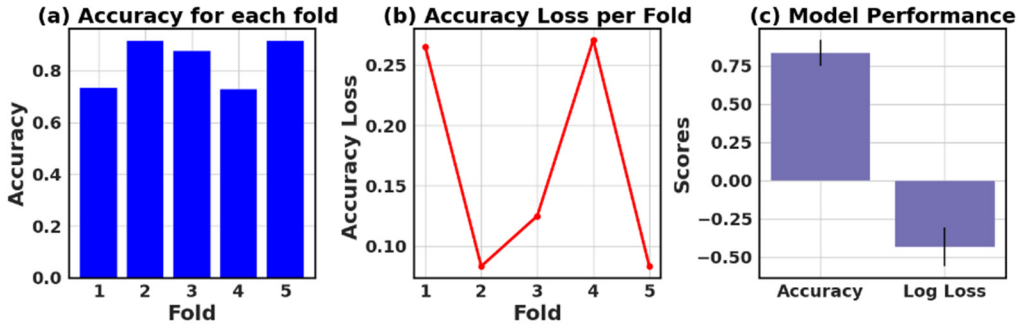


**Figure 20.** LR model's 5-fold accuracy and loss value plots for Dataset I.

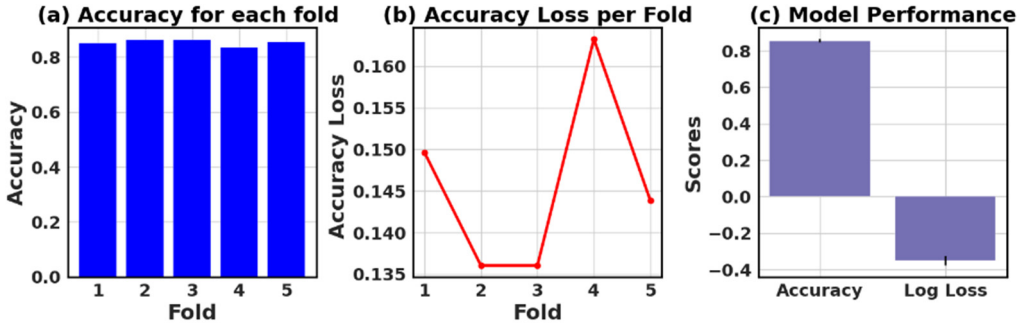


**Figure 21.** LR model's 5-fold accuracy and loss value plots for Dataset II.

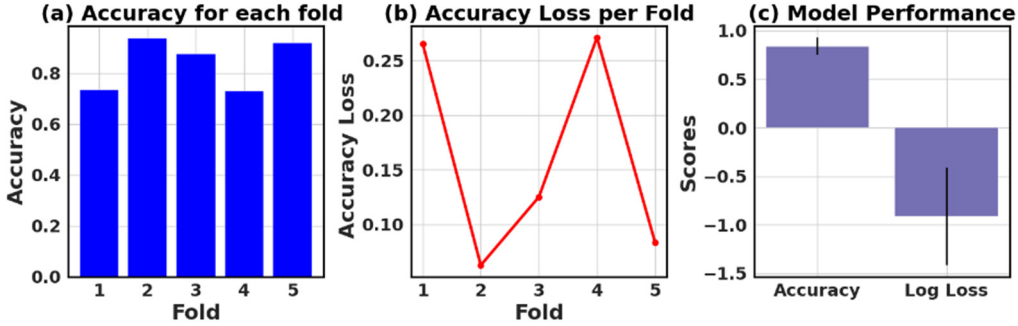


**Figure 22.** NB model's 5-fold accuracy and loss value plots for Dataset I.

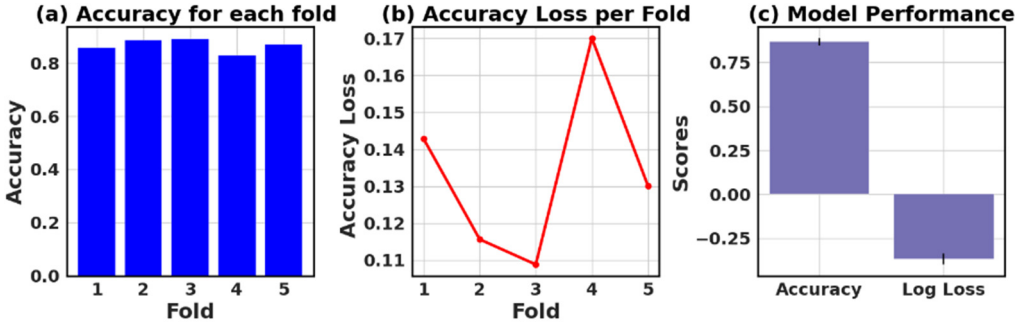


**Figure 23.** NB model's 5-fold accuracy and loss value plots for Dataset II.

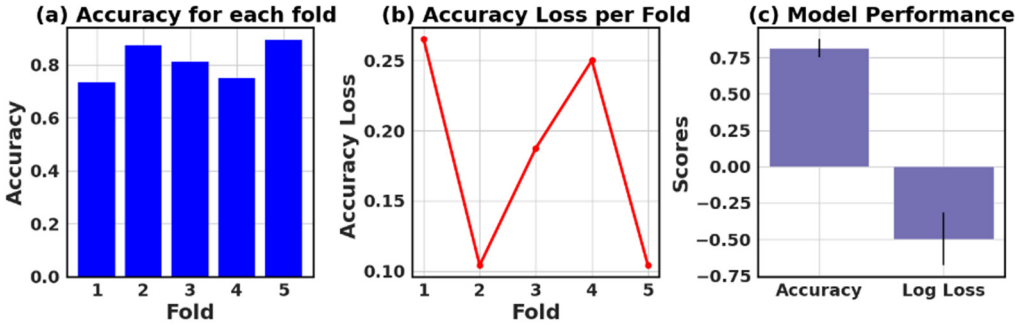


**Figure 24.** GB model's 5-fold accuracy and loss value plots for Dataset I.

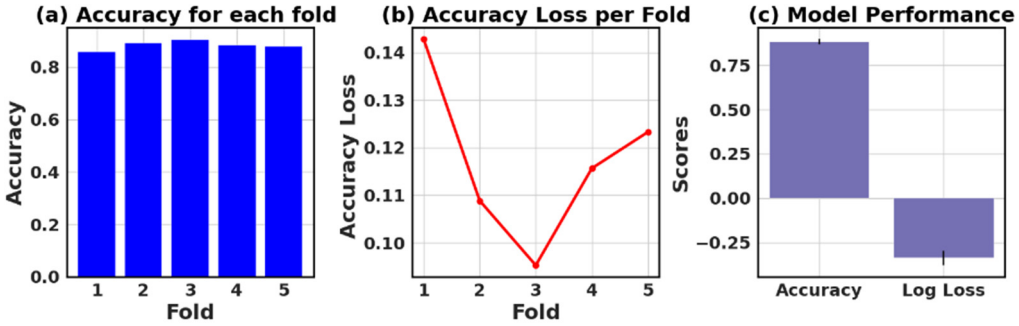


**Figure 25.** GB model's 5-fold accuracy and loss value plots for Dataset II.

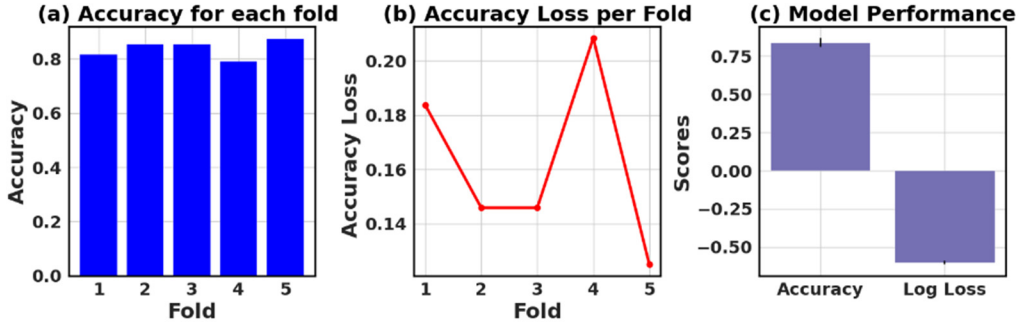


**Figure 26.** AB model's 5-fold accuracy and loss value plots for Dataset I.

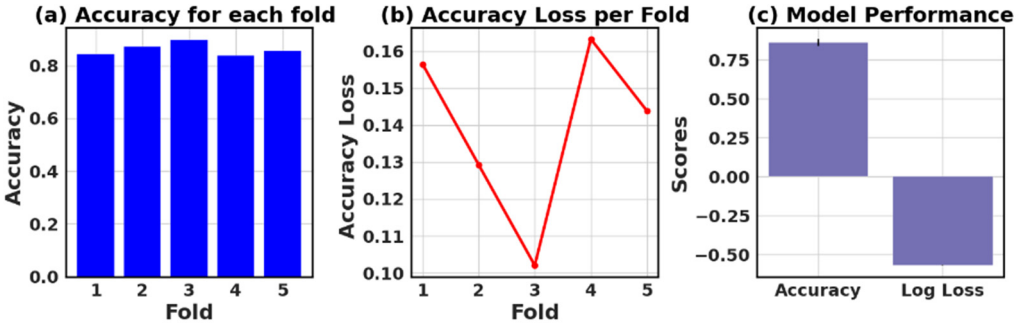


**Figure 27.** AB model's 5-fold accuracy and loss value plots for Dataset II.

**Table 12.** Different performance measure curve analysis values on ML models.

| Classifiers | Dataset | ROC-AUC (%) | Learning Curve (%) Average | | Precision Recall (%) |
|---|---|---|---|---|---|
| | | ROC of Class True, AUC | Training Score | CV Score | Average |
| RF | I | 95 | 88 | 76 | 96 |
| | II | 95 | 99 | 87 | 95 |
| KNN | I | 91 | 81 | 76 | 88 |
| | II | 93 | 86 | 84 | 93 |
| LR | I | 95 | 87 | 80 | 95 |
| | II | 94 | 87 | 84 | 95 |
| NB | I | 94 | 83 | 82 | 95 |
| | II | 94 | 87 | 85 | 94 |
| GB | I | 91 | 100 | 75 | 92 |
| | II | 94 | 99 | 84 | 94 |
| AB | I | 94 | 90 | 78 | 95 |
| | II | 95 | 94 | 80 | 95 |

**Table 13.** The values of accuracy, accuracy loss of each fold and mean and standard deviation values of six models.

| Model | Grid Search CV (5-Fold) Cross-Validation, Accuracy, Loss of Each Fold | | | | | | |
|---|---|---|---|---|---|---|---|
| | Dataset | | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 | Mean Value with a Standard Deviation |
| RF | I | Accuracy | 0.755 | 0.917 | 0.875 | 0.688 | 0.896 | 0.826 (+/−0.089) |
| | | Loss | 0.244 | 0.083 | 0.125 | 0.312 | 0.104 | 0.173 |
| | | Neg log loss | −0.48 | −0.411 | −0.375 | −0.56 | −0.343 | −0.434 (+/−0.078) |
| | II | Accuracy | 0.857 | 0.891 | 0.878 | 0.850 | 0.884 | 0.862 (+/−0.013) |
| | | Loss | 0.142 | 0.108 | 0.122 | 0.149 | 0.116 | 0.129 |
| | | Neg log loss | −0.359 | −0.308 | −0.308 | −0.361 | −0.351 | −0.337 (+/−0.028) |
| KNN | I | Accuracy | 0.755 | 0.917 | 0.833 | 0.688 | 0.917 | 0.822 (+/−0.090) |
| | | Loss | 0.244 | 0.083 | 0.166 | 0.312 | 0.083 | 0.178 |
| | | Neg log loss | −0.571 | −1.771 | −0.353 | −2.036 | −0.314 | −1.009 (+/−0.740) |
| | II | Accuracy | 0.857 | 0.857 | 0.871 | 0.844 | 0.877 | 0.863 (+/−0.012) |
| | | Loss | 0.142 | 0.142 | 0.129 | 0.156 | 0.123 | 0.142 |
| | | Neg log loss | −0.383 | −0.302 | −0.281 | −0.337 | −0.347 | −0.330 (+/−0.036) |
| LR | I | Accuracy | 0.735 | 0.917 | 0.875 | 0.729 | 0.917 | 0.834 (+/−0.085) |
| | | Loss | 0.265 | 0.083 | 0.125 | 0.270 | 0.083 | 0.165 |
| | | Neg log loss | −0.553 | −0.354 | −0.342 | 0.618 | −0.301 | −0.434 (+/−0.127) |
| | II | Accuracy | 0.850 | 0.864 | 0.864 | 0.837 | 0.856 | 0.854 (+/−0.010) |
| | | Loss | 0.149 | 0.136 | 0.136 | 0.163 | 0.143 | 0.145 |
| | | Neg log loss | −0.347 | −0.336 | −0.311 | −0.381 | −0.381 | −0.351 (+/−0.027) |
| NB | I | Accuracy | 0.735 | 0.938 | 0.875 | 0.729 | 0.917 | 0.839 (+/−0.089) |
| | | Loss | 0.265 | 0.062 | 0.125 | 0.270 | 0.083 | 0.161 |
| | | Neg log loss | −1.29 | −0.437 | −0.7 | −1.702 | −0.434 | −0.913 (+/−0.503) |
| | II | Accuracy | 0.857 | 0.884 | 0.891 | 0.830 | 0.870 | 0.866 (+/−0.022) |
| | | Loss | 0.142 | 0.115 | 0.108 | 0.170 | 0.130 | 0.132 |
| | | Neg log loss | −0.369 | −0.328 | −0.333 | −0.412 | −0.391 | −0.367 (+/−0.032) |
| GB | I | Accuracy | 0.735 | 0.875 | 0.854 | 0.771 | 0.896 | 0.826 (+/−0.060) |
| | | Loss | 0.265 | 0.104 | 0.145 | 0.229 | 0.104 | 0.173 |
| | | Neg log loss | −0.768 | −0.372 | −0.41 | −0.68 | −0.33 | −0.512 (+/−0.183) |
| | II | Accuracy | 0.857 | 0.891 | 0.905 | 0.884 | 0.877 | 0.887 (+/−0.016) |
| | | Loss | 0.142 | 0.108 | 0.095 | 0.115 | 0.123 | 0.115 |
| | | Neg log loss | −0.394 | −0.324 | −0.27 | −0.35 | −0.359 | −0.340 (+/−0.041) |
| AB | I | Accuracy | 0.816 | 0.854 | 0.854 | 0.792 | 0.875 | 0.838 (+/−0.030) |
| | | Loss | 0.183 | 0.145 | 0.145 | 0.208 | 0.125 | 0.161 |
| | | Neg log loss | −0.608 | −0.609 | −0.599 | −0.612 | −0.581 | −0.602 (+/−0.011) |
| | II | Accuracy | 0.844 | 0.871 | 0.898 | 0.837 | 0.856 | 0.862 (+/−0.022) |
| | | Loss | 0.156 | 0.129 | 0.102 | 0.163 | 0.143 | 0.102 |
| | | Neg log loss | −0.565 | −0.566 | −0.57 | −0.566 | −0.567 | −0.567 (+/−0.002) |

In this study, the performance of various ML models, including RF, KNN, LR, NB, GB, and AB, was evaluated on two different datasets (I and II). The models' performance was

assessed using five-fold cross-validation, and three metrics were reported: accuracy, loss (1—accuracy), and negative log loss.

The results indicate that, for Dataset I, the RF model achieved the highest mean accuracy (0.826) with a standard deviation of 0.089, followed closely by the NB and LR models with mean accuracies of 0.839 and 0.834, respectively. On the other hand, the KNN model had the lowest negative log loss (−1.009) with the largest standard deviation (0.740), which could suggest overfitting or instability in model performance across different folds.

For Dataset II, the GB showed the best performance with a mean accuracy of 0.887 and a standard deviation of 0.016. The other models, including KNN, LR, and NB, also demonstrated relatively high mean accuracies, ranging between 0.854 and 0.866. The negative log loss values were more stable for this dataset, with the AB model having the most consistent performance, indicated by a mean negative log loss of −0.567 and a standard deviation of 0.002. Further, this study reveals that selecting the best model requires careful consideration of the evaluation metrics and their respective standard deviations.

## 5. Methodology of Ensemble Classifier

In this proposed methodology, Multiple ML models are combined using the ensemble method to produce a collective result that is more accurate than any of the individual algorithms. Voting ensembles combine the predictions of our six ML models based on voting. In the voting classifier, there are two types of votes. These are hard votes and soft votes:

- Hard: the estimator selects the class prediction most frequently among the ML base models as the final class prediction by a majority vote.
- Soft: the final class prediction is based on the average probability considered from all the ML base model predictions.

Soft voting can yield better results than hard voting as it "gives more weight" to the confident votes by being an average of the probabilities. Both weighted and mean majority voting are considered in the soft voting ensemble. The soft voting ensemble (SVE) combines the predictions of individual models and uses the strengths of each model to make a more accurate prediction. In addition, the SVE reduces the risk of overfitting and is more robust to outliers and errors in the data. It is important to note that the performance of a SVE will depend on the problem and data being analyzed. The choice of base models and how their predictions are combined can also greatly impact the ensemble's performance [57]. A target label of essential probability can be selected in this manner. By doing so, individual classifiers are compensated for their shortcomings. The central aim of ensemble methods is to decrease the amount of bias and variance in a model.

Based on the scores of all forecasts by the base classifiers, the SVE method was used in our study in order to classify heart disease predictions. According to the proposed SVE model, the highest scores class is taken and the scores predicted by each of the base ML classifiers are added [58]. The SVE model we propose predicts the category with the highest probability value. Equation (11) shows the average score of each base classifier.

$$SVE = \text{argmax}(\frac{1}{N} \times (\ P(RF) + P(KNN) + P(LR) + P(NB) + P(GB) + P(AB))) \quad (11)$$

where "$N$" denotes the number of base classifiers and "$P$" represents the probability of each base classifier and arg max (argument maximize) is the function that returns the class with the highest probability. Figure 28 illustrates an ensemble classifier model for soft voting.

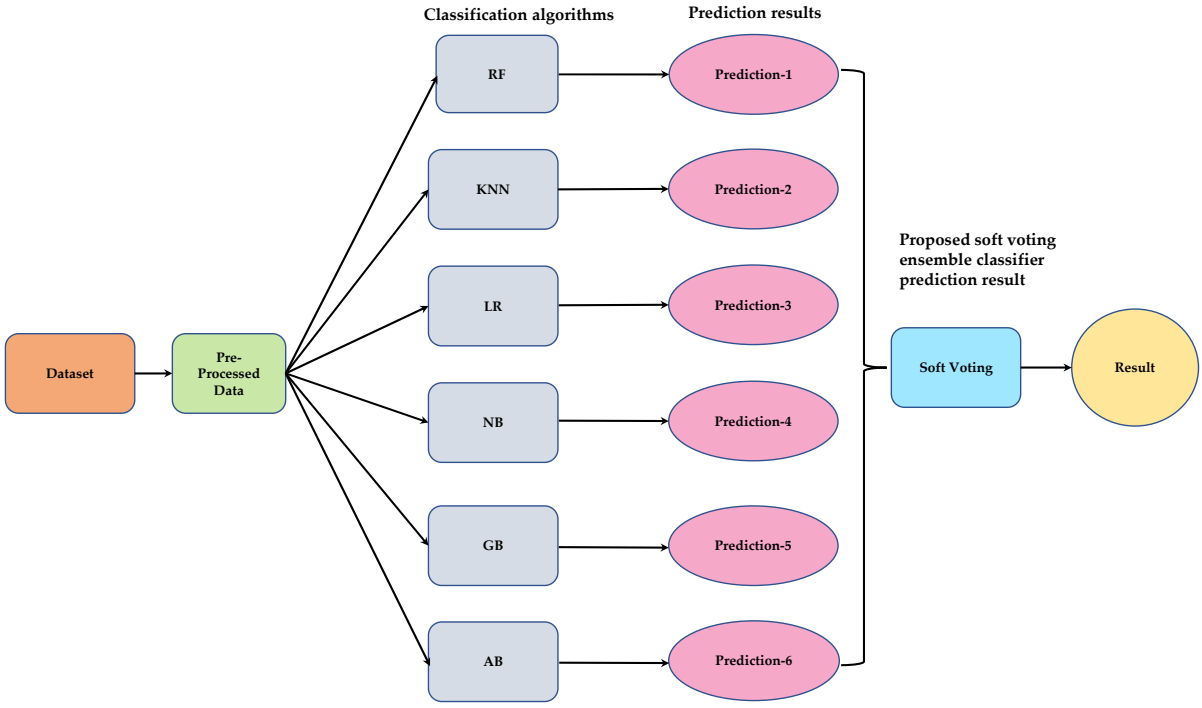


**Figure 28.** Proposed soft voting ensemble classifier.

## 6. Comparative Study

Figure 29 presents a performance analysis of six ML classifiers applied to Dataset I. The results show that LR achieves the highest accuracy of 90% among the classifiers, with notable precision and recall values for both classes. Other classifiers, such as RF, KNN, NB, GB, and AB display varying levels of performance across the different metrics. Upon examining the results, it is evident that the classifiers exhibit different strengths and weaknesses. For instance, while RF and AB have high precision for Class 0, they show lower recall values for the same class. Conversely, LR demonstrates remarkable precision for Class 0 and recall for Class 1.

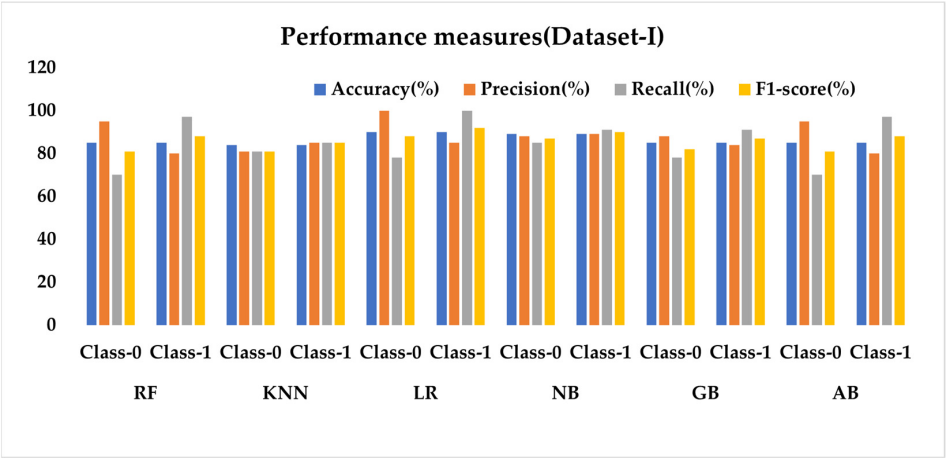


**Figure 29.** Performance measures comparison for six models on Dataset I.

Figure 30 represents the performance analysis for six ML classifiers. In this, the analysis reveals that the classifiers demonstrate relatively similar performance on Dataset II, with the AB classifier achieving the highest accuracy of 90%. Precision, recall, and F1 score values are also consistent across the classifiers. However, there are some differences

in performance, such as RF having a higher precision for Class 0 and a lower recall for the same class.

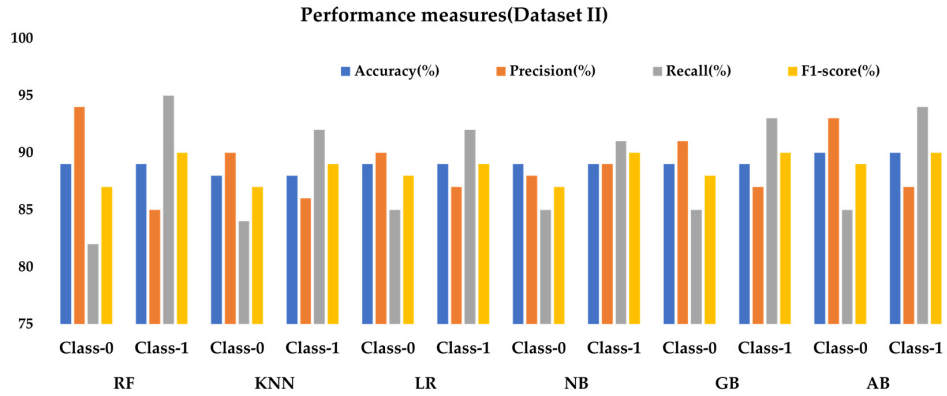


**Figure 30.** Performance measures comparison for six models on Dataset II.

Upon analyzing Figure 31, it is evident that the SVE classifier consistently outperforms the individual ML classifiers in both datasets, achieving 93.44% accuracy on Dataset I and 95% on Dataset II. As it is considered from individual classifiers, it has observed the maximum accuracy is only 90%, which is obtained from AB classifier on both the data sets. It is also notable that the performance of all classifiers improves from Dataset I to Dataset II. The SVE classifier effectively combines the strengths of the six individual classifiers, leading to enhanced accuracy in both datasets. This demonstrates the potential of ensemble methods for improved performance in heart disease prediction tasks. Tables 14 and 15 compare the previous researcher's accuracy and the proposed work result accuracy on Dataset I and Dataset II. Compared to the previous work results, our proposed model produced more accuracy.

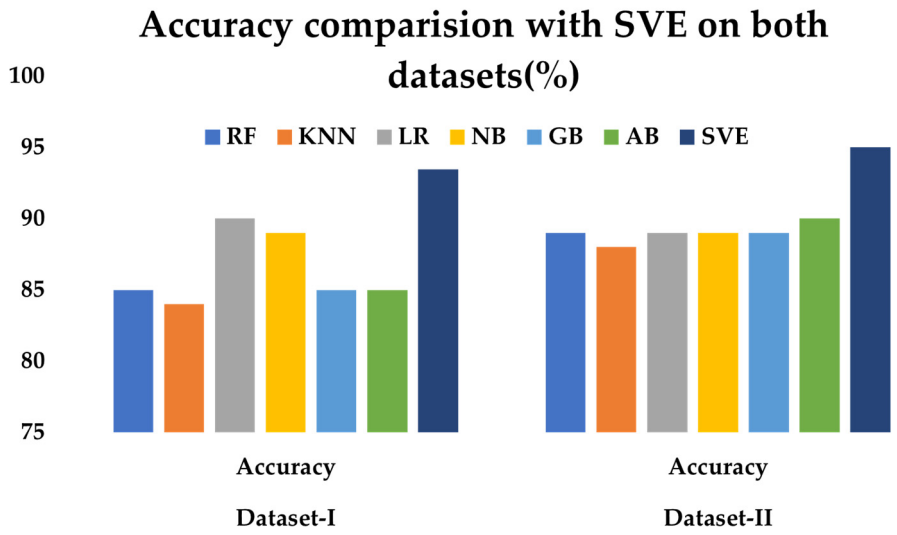


**Figure 31.** Proposed ensemble classifier accuracy compared with other six ML classifiers (Dataset I and Dataset II).

**Table 14.** Comparison of the proposed system with existing heart disease prediction systems on Dataset I.

| Ref. | Year | Dataset | Classifiers Used | Methodology Used | Maximum Accuracy (%) |
|---|---|---|---|---|---|
| [28]. | 2019 | Cleveland | NB Net, C 4.5, MLP, PART, Bagging, Boosting, majority voting, Stacking | Ensemble techniques such as bagging and boosting are employed for improving prediction accuracy. | 85.48 |
| [27]. | 2020 | Cleveland | LR, SVM, KNN | Feature normalization and dimensionality reduction utilizing principal component analysis (PCA) | 87.00 |
| [16]. | 2020 | Cleveland | LR, DT, and Gaussian naïve Bayes (GNB), | Dimensionality reduction was executed through singular value decomposition. | 82.75 |
| [34]. | 2022 | Cleveland | Stochastic Gradient Descent Classifiers, LR, SVM, NB, ConvSGLV, and Ensemble methods | Majority voting, CNN has been utilized for feature extraction with flatten layer converting 3D data into 1D as ML models work on 1D data. | 93.00 |
| [59] | 2023 | Cleveland | LR, KNN, DT, XGB, SVM, RF | GridsearchCV hyperparameter tuning. | 87.91 |
| **Proposed** | | **Cleveland dataset** | **RF, KNN, LR, NB, GB, AB, SVE classifier** | **Soft voting ensemble method.** | **93.44** |

**Table 15.** Comparison of the proposed system with existing heart disease prediction systems on Dataset II.

| Ref. | Year | Dataset | Classifiers Used | Methodology Used | Maximum Accuracy (%) |
|---|---|---|---|---|---|
| [38] | 2022 | Heart disease dataset (IEEE Dataport) | CART | Classification and regression tree algorithm. | 87.00 |
| [39] | 2021 | Heart disease dataset (IEEE Dataport) | RF, LR, SVM | A 10-fold repeated cross-validation method was employed. | 92.00 |
| [40] | 2022 | Heart disease dataset (IEEE Dataport) | NN, MLPNN, AB, SVM, LR, ANN, RF | An ensemble strategy that combines multiple classifiers. | 93.39 |
| **Proposed** | | **Heart disease dataset (IEEE Dataport)** | **RF, KNN, LR, NB, GB, AB, SVE classifier** | **Soft voting ensemble method.** | **95.00** |

The limitation of this model is that it is based on a limited amount of patient data, which only include 303 and 1190 patients in the datasets. Future work includes more patient data, the application of the feature selection method, and the development of a deep learning-based system for early heart disease detection. Additionally, utilizing medical IoT devices and sensors for the simultaneous collection of clinical parameters such as ECG, blood oxygen level, and body temperature can further improve the performance of the proposed system.

## 7. Conclusions

In conclusion, this research presents an efficient ML-based diagnosis system for detecting heart disease. To get the best accuracy results, the GridsearchCV hyperparameter method and the five-fold cross-validation method have been used before implementing models. Six ML classifiers were implemented and compared using accuracy, precision, recall, and F1 score metrics. The results indicate that the LR and AB classifiers attained the highest accuracies of 90.16% and 89.67% on both datasets, respectively. However, when the soft voting ensemble classifier method was applied to all six models on both datasets, it yielded even greater accuracies of 93.44% and 95%. To use this ML model for real-time heart disease prediction, it is necessary to integrate the model into a practical application. This can be achieved through a web application, mobile app, or other software systems. By deploying the model in a real-world setting, such as a hospital or clinic, it can be used to predict heart disease risk for patients. The model can also be integrated

into an electronic health record (EHR) system and make use of the patient's EHR data for real-time predictions.

**Author Contributions:** Conceptualization, N.C.; methodology, N.C.; software, N.C.; validation, N.C. and S.P.; formal analysis, N.C.; investigation, N.C.; resources, N.C.; data curation, N.C.; writing—original draft preparation, N.C.; writing—review and editing, S.P.; visualization, S.P.; supervision, S.P.; project administration, S.P.; funding acquisition, S.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The data presented in this study are available in the public database. Heart Dataset and Heart Attack Dataset. Retrieved from https://archive.ics.uci.edu/ml/datasets/Heart+Disease (accessed on 10 December 2022) and https://ieee-dataport.org/open-access/heart-disease-dataset-comprehensive (accessed on 12 November 2022).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. World Health Statistics. Cardiovascular Diseases, Key Facts. 2021. Available online: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds) (accessed on 10 December 2022).
2. Choudhury, R.P.; Akbar, N. Beyond Diabetes: A Relationship between Cardiovascular Outcomes and Glycaemic Index. *Cardiovasc. Res.* **2021**, *117*, E97–E98. [CrossRef] [PubMed]
3. Ordonez, C. Association Rule Discovery with the Train and Test Approach for Heart Disease Prediction. *IEEE Trans. Inf. Technol. Biomed.* **2006**, *10*, 334–343. [CrossRef] [PubMed]
4. Magesh, G.; Swarnalatha, P. Optimal Feature Selection through a Cluster-Based DT Learning (CDTL) in Heart Disease Prediction. *Evol. Intell.* **2021**, *14*, 583–593. [CrossRef]
5. Rohit Chowdary, K.; Bhargav, P.; Nikhil, N.; Varun, K.; Jayanthi, D. Early Heart Disease Prediction Using Ensemble Learning Techniques. *J. Phys. Conf. Ser.* **2022**, *2325*, 012051. [CrossRef]
6. Liu, J.; Dong, X.; Zhao, H.; Tian, Y. Predictive Classifier for Cardiovascular Disease Based on Stacking Model Fusion. *Processes* **2022**, *10*, 749. [CrossRef]
7. Devi, A.G. A Method of Cardiovascular Disease Prediction Using Machine Learning. *Int. J. Eng. Res. Technol.* **2021**, *9*, 243–246.
8. Uddin, S.; Khan, A.; Hossain, M.E.; Moni, M.A. Comparing Different Supervised Machine Learning Algorithms for Disease Prediction. *BMC Med. Inform. Decis. Mak.* **2019**, *19*, 281. [CrossRef]
9. Patro, S.P.; Nayak, G.S.; Padhy, N. Heart Disease Prediction by Using Novel Optimization Algorithm: A Supervised Learning Prospective. *Inform. Med. Unlocked* **2021**, *26*, 100696. [CrossRef]
10. Song, Q.; Zheng, Y.J.; Yang, J. Effects of Food Contamination on Gastrointestinal Morbidity: Comparison of Different Machine-Learning Methods. *Int. J. Environ. Res. Public Health* **2019**, *16*, 838. [CrossRef]
11. Pasha, S.J.; Mohamed, E.S. Novel Feature Reduction (NFR) Model with Machine Learning and Data Mining Algorithms for Effective Disease Risk Prediction. *IEEE Access* **2020**, *8*, 184087–184108. [CrossRef]
12. Gupta, A.; Kumar, R.; Singh Arora, H.; Raman, B. MIFH: A Machine Intelligence Framework for Heart Disease Diagnosis. *IEEE Access* **2020**, *8*, 14659–14674. [CrossRef]
13. Rani, P.; Kumar, R.; Ahmed, N.M.O.S.; Jain, A. A Decision Support System for Heart Disease Prediction Based upon Machine Learning. *J. Reliab. Intell. Environ.* **2021**, *7*, 263–275. [CrossRef]
14. Jordanov, I.; Petrov, N.; Petrozziello, A. Classifiers Accuracy Improvement Based on Missing Data Imputation. *J. Artif. Intell. Soft Comput. Res.* **2018**, *8*, 31–48. [CrossRef]
15. Ambrish, G.; Ganesh, B.; Ganesh, A.; Srinivas, C.; Mensinkal, K. Logistic Regression Technique for Prediction of Cardiovascular Disease. *Glob. Transit. Proc.* **2022**, *3*, 127–130. [CrossRef]
16. Ananey-Obiri, D.; Sarku, E. Predicting the Presence of Heart Diseases Using Comparative Data Mining and Machine Learning Algorithms. *Int. J. Comput. Appl.* **2020**, *176*, 17–21. [CrossRef]
17. Mohan, S.; Thirumalai, C.; Srivastava, G. Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques. *IEEE Access* **2019**, *7*, 81542–81554. [CrossRef]
18. Kodati, S.; Vivekanandam, R. Analysis of Heart Disease Using in Data Mining Tools Orange and Weka Sri Satya Sai University Analysis of Heart Disease Using in Data Mining Tools Orange and Weka. *Glob. J. Comput. Sci. Technol. C* **2018**, *18*, 17–21.
19. Shah, S.M.S.; Batool, S.; Khan, I.; Ashraf, M.U.; Abbas, S.H.; Hussain, S.A. Feature Extraction through Parallel Probabilistic Principal Component Analysis for Heart Disease Diagnosis. *Phys. A Stat. Mech. Its Appl.* **2017**, *482*, 796–807. [CrossRef]
20. Perumal, R. Early Prediction of Coronary Heart Disease from Cleveland Dataset Using Machine Learning Techniques. *Int. J. Adv. Sci. Technol.* **2020**, *29*, 4225–4234.
21. Vijayashree, J.; Sultana, H.P. A Machine Learning Framework for Feature Selection in Heart Disease Classification Using Improved Particle Swarm Optimization with Support Vector Machine Classifier. *Program. Comput. Softw.* **2018**, *44*, 388–397. [CrossRef]

22. Yekkala, I.; Dixit, S. Prediction of Heart Disease Using Random Forest and Rough Set Based Feature Selection. *Int. J. Big Data Anal. Healthc.* **2018**, *3*, 12. [CrossRef]
23. Saw, M.; Saxena, T.; Kaithwas, S.; Yadav, R.; Lal, N. Estimation of Prediction for Getting Heart Disease Using Logistic Regression Model of Machine Learning. In Proceedings of the 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 22–24 January 2020. [CrossRef]
24. Otoom, A.F.; Abdallah, E.E.; Kilani, Y.; Kefaye, A. Effective Diagnosis and Monitoring of Heart Disease. *Int. J. Softw. Eng. Its Appl.* **2015**, *9*, 143–156.
25. Vembandasamy, K.; Sasipriya, R.; Deepa, E. Heart Diseases Detection Using Naive Bayes Algorithm. *Int. J. Innov. Sci. Eng. Technol.* **2015**, *2*, 441–444.
26. Gazeloğlu, C. Prediction of Heart Disease by Classifying with Feature Selection and Machine Learning Methods. *Prog. Nutr.* **2020**, *22*, 660–670. [CrossRef]
27. Reddy, K.V.V.; Elamvazuthi, I.; Aziz, A.A.; Paramasivam, S.; Chua, H.N.; Pranavanand, S. Heart Disease Risk Prediction Using Machine Learning Classifiers with Attribute Evaluators. *Appl. Sci.* **2021**, *11*, 8352. [CrossRef]
28. Pavithra, V.; Jayalakshmi, V. Hybrid Feature Selection Technique for Prediction of Cardiovascular Diseases. *Mater. Today Proc.* 2021; *in press*. [CrossRef]
29. Latha, C.B.C.; Jeeva, S.C. Improving the Accuracy of Prediction of Heart Disease Risk Based on Ensemble Classification Techniques. *Inform. Med. Unlocked* **2019**, *16*, 100203. [CrossRef]
30. Bashir, S.; Qamar, U.; Khan, F.H.; Javed, M.Y. MV5: A Clinical Decision Support Framework for Heart Disease Prediction Using Majority Vote Based Classifier Ensemble. *Arab. J. Sci. Eng.* **2014**, *39*, 7771–7783. [CrossRef]
31. Tama, B.A.; Im, S.; Lee, S. Improving an Intelligent Detection System for Coronary Heart Disease Using a Two-Tier Classifier Ensemble. *BioMed Res. Int.* **2020**, *2020*, 9816142. [CrossRef]
32. Alqahtani, A.; Alsubai, S.; Sha, M.; Vilcekova, L.; Javed, T. Cardiovascular Disease Detection Using Ensemble Learning. *Comput. Intell. Neurosci.* **2022**, *2022*, 5267498. [CrossRef]
33. Trigka, M.; Dritsas, E. Long-Term Coronary Artery Disease Risk Prediction with Machine Learning Models. *Sensors* **2023**, *23*, 1193. [CrossRef] [PubMed]
34. Rustam, F.; Ishaq, A.; Munir, K.; Almutairi, M.; Aslam, N.; Ashraf, I. Incorporating CNN Features for Optimizing Performance of Ensemble Classifier for Cardiovascular Disease Prediction. *Diagnostics* **2022**, *12*, 1474. [CrossRef] [PubMed]
35. Cyriac, S.; Sivakumar, R.; Raju, N.; Woon Kim, Y. Heart Disease Prediction Using Ensemble Voting Methods in Machine Learning. In Proceedings of the 2022 13th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 19–21 October 2022; pp. 1326–1331. [CrossRef]
36. Jan, M.; Awan, A.A.; Khalid, M.S.; Nisar, S. Ensemble Approach for Developing a Smart Heart Disease Prediction System Using Classification Algorithms. *Res. Rep. Clin. Cardiol.* **2018**, *9*, 33–45. [CrossRef]
37. Manu Siddhartha Heart Disease Dataset (Comprehensive). Available online: https://ieee-dataport.org/authors/manu-siddhartha (accessed on 12 November 2022).
38. Ozcan, M.; Peker, S. A Classification and Regression Tree Algorithm for Heart Disease Modeling and Prediction. *Healthc. Anal.* **2023**, *3*, 100130. [CrossRef]
39. Yilmaz, R.; Yağin, F.H. Early Detection of Coronary Heart Disease Based on Machine Learning Methods. *Med. Rec.* **2021**, *4*, 1–6. [CrossRef]
40. Doppala, B.P.; Bhattacharyya, D.; Janarthanan, M.; Baik, N. A Reliable Machine Intelligence Model for Accurate Identification of Cardiovascular Diseases Using Ensemble Techniques. *J. Healthc. Eng.* **2022**, *2022*, 2585235. [CrossRef]
41. UCI Machine Learning Repository Heart Disease Data Set. Available online: https://archive.ics.uci.edu/ml/datasets/Heart+Disease (accessed on 10 December 2022).
42. IEEE Dataport Heart Disease Dataset. Available online: https://ieee-dataport.org/open-access/heart-disease-dataset-comprehensive (accessed on 12 November 2022).
43. Bharti, R.; Khamparia, A.; Shabaz, M.; Dhiman, G.; Pande, S.; Singh, P. Prediction of Heart Disease Using a Combination of Machine Learning and Deep Learning. *Comput. Intell. Neurosci.* **2021**, *2021*, 8387680. [CrossRef]
44. Kumari, M.; Ahlawat, P. DCPM: An Effective and Robust Approach for Diabetes Classification and Prediction. *Int. J. Inf. Technol.* **2021**, *13*, 1079–1088. [CrossRef]
45. Biswas, P.; Samanta, T. Anomaly Detection Using Ensemble Random Forest in Wireless Sensor Network. *Int. J. Inf. Technol.* **2021**, *13*, 2043–2052. [CrossRef]
46. Sengupta, S.; Mayya, V.; Kamath, S.S. Detection of Bradycardia from Electrocardiogram Signals Using Feature Extraction and Snapshot Ensembling. *Int. J. Inf. Technol.* **2022**, *14*, 3235–3244. [CrossRef]
47. Sahu, A.; Gm, H.; Gourisaria, M.K.; Rautaray, S.S.; Pandey, M. Cardiovascular Risk Assessment Using Data Mining Inferencing and Feature Engineering Techniques. *Int. J. Inf. Technol.* **2021**, *13*, 2011–2023. [CrossRef]
48. Saqlain, M.; Jargalsaikhan, B.; Lee, J.Y. A Voting Ensemble Classifier for Wafer Map Defect Patterns Identification in Semiconductor Manufacturing. *IEEE Trans. Semicond. Manuf.* **2019**, *32*, 171–182. [CrossRef]
49. Miao, J.; Zhu, W. Precision–Recall Curve (PRC) Classification Trees. *Evol. Intell.* **2022**, *15*, 1545–1569. [CrossRef]
50. Pal, M.; Parija, S. Prediction of Heart Diseases Using Random Forest. *J. Phys. Conf. Ser.* **2021**, *1817*, 012009. [CrossRef]

51. Polat, K.; Güneş, S. A New Feature Selection Method on Classification of Medical Datasets: Kernel F-Score Feature Selection. *Expert Syst. Appl.* **2009**, *36*, 10367–10373. [CrossRef]
52. Verma, P. Ensemble Models for Classification of Coronary Artery Disease Using Decision Trees. *Int. J. Recent Technol. Eng.* **2020**, *8*, 940–944. [CrossRef]
53. Sharma, A.; Mishra, P.K. Performance Analysis of Machine Learning Based Optimized Feature Selection Approaches for Breast Cancer Diagnosis. *Int. J. Inf. Technol.* **2022**, *14*, 1949–1960. [CrossRef]
54. Sarwar, A.; Ali, M.; Manhas, J.; Sharma, V. Diagnosis of Diabetes Type-II Using Hybrid Machine Learning Based Ensemble Model. *Int. J. Inf. Technol.* **2020**, *12*, 419–428. [CrossRef]
55. Al Bataineh, A.; Manacek, S. MLP-PSO Hybrid Algorithm for Heart Disease Prediction. *J. Pers. Med.* **2022**, *12*, 1208. [CrossRef]
56. Guleria, P.; Naga Srinivasu, P.; Ahmed, S.; Almusallam, N.; Alarfaj, F.K. XAI Framework for Cardiovascular Disease Prediction Using Classification Techniques. *Electronics* **2022**, *11*, 4086. [CrossRef]
57. Ali, S.; Hussain, A.; Aich, S.; Park, M.S.; Chung, M.P.; Jeong, S.H.; Song, J.W.; Lee, J.H.; Kim, H.C. A Soft Voting Ensemble-Based Model for the Early Prediction of Idiopathic Pulmonary Fibrosis (IPF) Disease Severity in Lungs Disease Patients. *Life* **2021**, *11*, 1092. [CrossRef] [PubMed]
58. Manconi, A.; Armano, G.; Gnocchi, M.; Milanesi, L. A Soft-Voting Ensemble Classifier for Detecting Patients Affected by COVID-19. *Appl. Sci.* **2022**, *12*, 7554. [CrossRef]
59. Ahamad, G.N.; Fatima, H.; Zakariya, S.M.; Abbas, M. Influence of Optimal Hyperparameters on the Performance of Machine Learning Algorithms for Predicting Heart Disease. *Processes* **2023**, *11*, 734. [CrossRef]